RECOGNITION OF FIDUCIAL MARKS APPLIED TO ROBOTIC SYSTEMS

NAGW-1333

by

Wayne D. Georges

Rensselaer Polytechnic Institute
Electrical, Computer, and Systems Engineering Department
Troy, New York 12180-3590

August 1991

CIRSSE REPORT #103

CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENT

I would like to thank my mother, without whose numerous sacrifices and words of encouragement, I would have never made it this far. I would like to thank my father for his constant belief in me. I would like to sincerely thank my entire family and my friends for their support over the years. I would also like to thank my thesis advisor Dr. Robert Kelley, for his valuable guidance during the course of this project. Finally, I would like to thank Sharon for her help and support when it was needed the most.

ABSTRACT

The objective of this project is to devise a method to determine the position and orientation of the links of a PUMA 560 using fiducial marks, As a result, it is necessary to design fiducial marks and a corresponding feature extraction algorithm. The marks utilized are composites of three basic shapes, a circle, an equilateral triangle and a square.

Once a mark is imaged it is thresholded and the borders of each shape are extracted. These borders are subsequently utilized in a feature extraction algorithm. Two feature extraction algorithms are utilized to determine which one produces the most reliable results. The first algorithm is based on moment invariants and the second algorithm is based on the discrete version of the ψ - s curve of the boundary. The latter algorithm is clearly superior for this application.

CHAPTER 1
INTRODUCTION

1.1 Overview

Computer vision is an essential part of any intelligent robotic system. It serves as a means to identify the environment and to verify the robot's location and orientation. This information provides the parameters to determine the proper action to be taken by the system.

In the verification of the robot's position and orientation, a means must be found to identify the position and orientation of each joint. The use of fiducial marks on each link seems to be an effective and cost efficient way to accomplish this task. The marks provide a way of uniquely identifying each link and locating a point on the link. This point along with two calibrated CCD cameras can determine the 3D location.

There are several criteria the marks must satisfy to be effective tools. They must be simple enough so that they are identifiable even under conditions of low resolution, perspective distortion, rotations, scale changes, and translations. The above can be stated in another way. There must exist an algorithm that can extract features from the marks that are rotation, translation, and scale invariant, and the features should be as insensitive as possible to perspective distortion. For a discrete image it is not possible to have features that are truly rotationally and scale invariant. In reality the features are a function of the resolution of the digitizing grid. Features that satisfy the above constraints are sufficient for the task of determining the status of the link. The position and orientation of a link can be modeled, in three dimensional space, as an ordered rotation about the x, y, and z axis and a translation. This transformation, the link and its corresponding mark are subjected

to a projective transformation which is a noninvertible transformation of three-dimensional world space into a two-dimensional image plane. Therefore, every point in the image is a function of its positon in world coordinates, the focal length of the CCD camera, and the field of view of the camera. In addition, the image of the fiducial mark is a function of the distribution of the points around its centroid, the three rotation angles of the plane that the marks lie upon, the position of the centroid, the focal length and the field of view.

Moments provide an excellent way to characterize mass distributions: such as horizontal and vertical centralness, diagonality, horizontal and vertical divergence, and horizontal and vertical imbalance. Another convenient feature of moments is their ability to be normalized for scale changes and rotations and translations in the image plane. Many of these invariants can be obtained by using either the theory of algebric invariants, introduced by Cayley, Hamilton, and Sylvester, or by requiring that certain lower order moments have a prescribed value, and normalizing the other moments with respect to these lower order moments. Another convenient feature of moments is their ease of calculation.

1.2 Literature Survey

Many papers have been written on the use of moments in pattern recognition applications. One of the first is the paper written by Hu[1]. In this paper Hu discusses recognition of two-dimensional geometric patterns by using the classical theory of algebric invariants to derive moment invariants that are insensitive to scale, position, and orientation. This method uses invariant moments based upon uniquely determined principal axes and the method of absolute moment invariants. These moment invariants are subsequently stored in a feature vector and compared, using a minimum distance formulation, to feature vectors of known patterns.

Udagawa et alia[2] use moments to identify capital letters of the English alphabet. Their method consists of normalizing linearly distorted patterns by setting certain conditions on the lower order moments. The method essentially normalizes for any distortion due to an affine transformation. The normalized moments are used as recognition features.

Alt[3] uses moments to identify letters and numbers. He normalizes each pattern with respect to position, size, stretching and squeezing in the x or y directions and slanting in the x direction. The patterns are not rotationally invariant. The rotational variance is done to facilitate the discrimination of 6's and 9's. The slant invariance is utilized to identify italics and bold faced letters as the same pattern. Normalization is accomplished by utilizing the standard deviation and the regression coefficient of x on y. Third through sixth order moments are calculated for the twenty-six capital letters and nine numbers. The discrimination algorithm searches for gaps in the values of a particular moment. These gaps are discrimination points that separate certain patterns from others. Once subregions are formed based on these points, another moment is used to break the subregions into smaller regions. The process continues until each subregion consists of one element.

Casey[4] deals with the problem of normalizing handprinted characters. Because of the large disparity in handwriting styles, recognition of characters is a difficult task. He models the distortion as an affine transformation. This information is used to direct the direction of scan of an optical character recognition device to obtain a more uniform scan of letters. He uses the same methodology as Udagawa.

Smith and Wright[5] uses the method of moments to estimate the location, orientation, length, width, and heading of a ship. The estimates are obtained by taking moments of a ship photograph and using linear, quadratic, and cubic polynomial functions of the moments as estimators of the ship descriptors. The best moments

for each polynomial are chosen using linear regression. This research verifies the feasibility of using moments to interpret ship photographs.

Dudani, Breeding, and McGhee[6] address the problem of aircraft identification. The images of the airplanes are binearized and moment invariants are extracted from the image. They preprocess the two-dimensional binary image of the three-dimensional aircraft and extract a clean silhouette and its corresponding boundary. The algorithm employed is orientation invariant. The dimensionality of the feature vector is kept as low as possible and is shown to be invariant to translation in the plane normal to the optical axis. The moment invariants employed are the Hu invariants divided by a power of the radius of gyration. They calculate two sets of moments, one for the silhouette and one for the boundary. The boundary moments are found to contain a large amount of information on the high frequency content of the image. To identify the images they created a recognizer that consisted of 3,000 live images of six types of aircrafts. These samples are obtained by imaging each aircraft at various orientations. They then map the feature space to a space defined by the set of eigenvectors corresponding to the training sample covariance matrix. The set of feature components is ordered according to the information content. Two types of decision rules are employed to classify unknown images, Bayes decision rule and the distance-weighted k-nearest neighbor algorithm. The results of the algorithms are compared to the decisions made by human observers. Both algorithms outperform the human observers, but each computer decision took thirty seconds whereas the human observers take between ten and fifteen seconds. The algorithm achieves reasonable accuracy in estimating the aircrafts inclination. The errors are typically between five and ten degrees.

Teague[7] addresses the issue of classifying and manipulating optical information by utilizing moments. He summarizes the properties of the lower order geometric moments. The merits of Zernike moments are addressed in relation to rotational

invariance and optimal reconstruction of an image. It is also shown that Zernike moments can be easily derived from the geometric moments. He demonstrates the advantage of using the Orthogonal moments in image reconstruction.

Wong and Hall[8] use geometric moment invariants to match radar images to their corresponding optical images. Because the invariants are calculated for continuous images these moments are not strictly invariant for digital images. The amount of discrepancy is a function of the amount of the scale, translation and rotation change. According to their data, reasonably good results can be obtained for rotations up to forty-five degrees and scale changes of less than a factor of two. They designed a hierarchical search technique to match the radar to the optical scenes. This scheme consists of extracting a structural set of images, both radar and optical, which are of decreasing size and resolution. The match sequence starts with the lower resolution images. A thresholding algorithm and decision rule is utilized to guide the search from a lower resolution level to a higher resolution level. The rules are selected to find the most promising locations at each level. Only these areas are tested at the next higher resolution. A product correlator is used to match the invariant moments of the radar subimages to their corresponding optical subimages.

Boyce and Hossack[9] construct feature vectors of arbitrary order while maintaining the significance of the higher order components of the feature vector. The features are Zernike moments and the rotational moments. Reconstruction of the images based upon a finite number of Zernike moments is discussed. The invariants used in the feature vector are rotational moments. The transformation is invariant to scale, intensity, rotation, and translation. The goal is to create features that are independent and are of approximately equal orders to magnitude. This insures that the information content is not overly sensitive to noise. The rotational moments are used to identify the image and the Zernike moments are used to reconstruct the

image.

Khotanzad and Hong[10] describe new rotationally invariant features using Zernike moments, and a systematic method to select the desired number of features. This is accomplished by evaluating the discrimination power of the information content of the *ith* ordered features of different classes. The patterns are grouped into pairs. The pairs are subsequently rotationally aligned, and the Hamming distance of the information content of the pair is taken. A cumulative measure of the Hamming distance is obtained, and this is divided by the total number of pixels. This value is divided by one more than the feature number to provide a measure of the discrimination power. When this discrimination power exceeds a preset threshold then the number of features needed is known for the pair. The maximum value of all the pairs is taken as the number of features needed for the given patterns.

1.3 Author's contribution

It is necessary to identify the position and orientation of the links of a robotic manipulator (PUMA 560). To accomplish this task several fiducial marks will be placed upon each link. What remains is to identify each fiducial mark (a pattern recognition problem) and to locate a point associated with each link (the centroid). The first portion of the problem consists of designing an adequate number of simple fiducial marks. This is done to label the links sufficiently and to facilitate the extraction of recognizable features under conditions of low resolution and perspective or orthogonal projection.

The marks employed are designed from simple shapes - such as circles, squares, and equilateral triangles. Since it is necessary to generate a large number of marks from these basic shapes, the idea of nesting shapes within shapes is introduced. Each composite pattern is designed such that each interior shape is completely contained in its parent shape, and each interior shape has a grey level intensity value that

contrasts with its parent shape. Using this methodology and two level of nesting, it is possible to generate twenty seven unique fiducial marks.

Since the twenty seven generated fiducial marks are composites of the three basic shapes, the extraction of the borders of each of the shapes contained within the mark reduces the recognition problem from one of extracting the features from twenty seven unique patterns to that of extracting features of three shapes. Once the borders of each of the interior shapes are identified, the results are combined to yield the correct identification of the mark.

Moments are extracted from each of the borders to determine whether they are reliable features. Since the moments are relatively simple to calculate, it is of interest to determine if they can be used to identify the shapes in binary, low resolution, perspective distorted images. It is also of interest to determine whether the normalized moments of orders two and three can be used to accomplish this recognition problem.

Features based on the ψ-s curves of the boundary are also used in this study. In particular, the measures of curvature obtained from these functions were used as features. These curvature measures are essentially local feature descriptors, and therefore are more susceptible to noisy border extractions and quantization effects. It is of interest to determine how these features perform under poor image conditions.

CHAPTER 2
MATHEMATICAL BACKGROUND

2.1 Rotation Matrices In Three-Dimensional Space

In many vision and or robotics applications, it is convenient to represent rotations of bodies or points around arbitrary axes in a convenient matrix form. Following [12], consider the derivation of the rotation matrix for rotations about the x-, y-, and z-axes (figure 2.1).

A convenient way to view this rotation is to consider two coordinate systems, XYZ and UVW centered at the origin and initially coincident.

Consider now a point P in this three-dimensional space. It has a representation in each of the coordinate systems denoted by

$$\begin{bmatrix} P_u \\ P_v \\ P_w \end{bmatrix} \quad and \quad \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix}$$

The point P is assumed to be rigidly attached to the UVW system. The goal is to find a rotation matrix that represents the rotation of the UVW coordinate system and the point P about the XYZ coordinate system.

The point P_{uvw} can be represented as a linear combination of the basis vectors of the UVW system.

$$P_{uvw} = P_u \mathrm{i}_u + P_v \mathrm{j}_v + P_w \mathrm{k}_w. \tag{2.1}$$

To obtain the mapping of P onto each of the basis vectors of the XYZ axes the dot product of P is taken with respect to the above basis vector. The results are as follows:

$$P_x = \mathrm{i}_x \bullet \mathrm{P} = \mathrm{i}_x \bullet \mathrm{i}_u P_u + \mathrm{i}_x \bullet \mathrm{j}_v P_v + \mathrm{i}_x \bullet \mathrm{k}_w P_w, \tag{2.2}$$

$$P_y = \mathrm{j}_y \bullet \mathrm{P} = \mathrm{j}_y \bullet \mathrm{i}_u P_u + \mathrm{j}_y \bullet \mathrm{j}_v P_v + \mathrm{j}_y \bullet \mathrm{k}_w P_w. \tag{2.3}$$

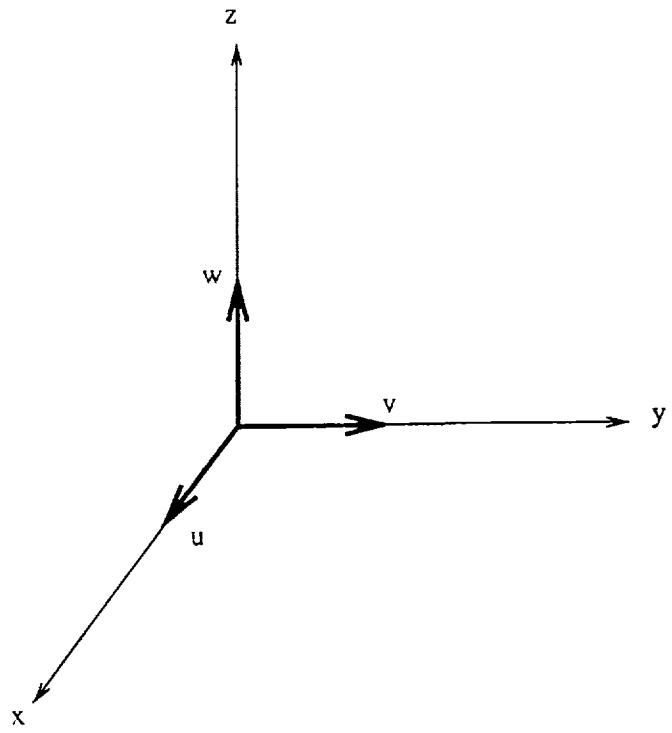

Figure 2.1: UVW and XYZ Coordinate Systems

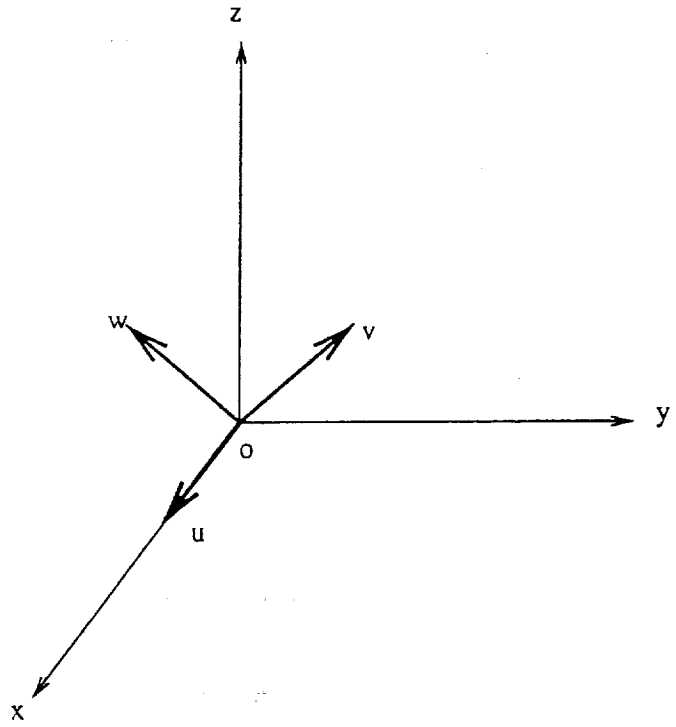

Figure 2.2: Rotation of the UVW coordinate system about the x axis

$$P_z = \mathrm{k}_z \bullet \mathrm{P} = \mathrm{k}_z \bullet \mathrm{i}_u P_u + \mathrm{k}_z \bullet \mathrm{j}_v P_v + \mathrm{k}_z \bullet \mathrm{k}_w P_w. \tag{2.4}$$

This can be expressed in matrix form as

$$\begin{pmatrix} P_x \\ P_y \\ P_z \end{pmatrix} = \begin{pmatrix} \mathrm{i}_x \bullet \mathrm{i}_x & \mathrm{i}_x \bullet \mathrm{j}_v & \mathrm{i}_x \bullet \mathrm{k}_w \\ \mathrm{j}_y \bullet \mathrm{i}_x & \mathrm{j}_y \bullet \mathrm{j}_v & \mathrm{j}_y \bullet \mathrm{k}_w \\ \mathrm{k}_z \bullet \mathrm{i}_x & \mathrm{k}_z \bullet \mathrm{j}_v & \mathrm{k}_z \bullet \mathrm{k}_w \end{pmatrix} \begin{pmatrix} P_u \\ P_v \\ P_w \end{pmatrix}$$

Keeping in mind that any rotation can be achieved by successive rotations about each of the three axes in the XYZ system, all that needs to be done is to obtain a matrix representation of rotations about each of the coordinate axes and then multiply the three matrices to obtain a composite rotation matrix. A rotation around the x axis by an angle α leaves the i_u axes fixed in relationship to the XYZ coordinate system (figure 2.2). Since the i_u axis is coincident with the i_x axes, and the j_v and k_w are rotated by an angle α with respect to the j_y and k_z axes

respectively, the following rotation matrix is obtained

$$R_{x,\alpha} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{pmatrix}.$$

The same procedure is followed for rotations about the y-axis to obtain the following matrix:

$$R_{y,\phi} = \begin{pmatrix} \cos\phi & 0 & sin\phi \\ 0 & 1 & 0 \\ \sin\phi & 0 & \cos\phi \end{pmatrix}.$$

Rotation about the z-axis is represented by:

$$R_{z,\theta} = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

A composite rotation matrix is obtained by multiplying the matrices together. Since matrix muplication is not commutative, the order of multiplication is important. For example, if one wanted to obtain the composite rotation matrix for a rotation about the z-axis by θ, followed by a rotation about the y-axis by ϕ, and then a rotation about the x-axis by α, the composite rotation matrix would be: $R = R_{x,\alpha}R_{y,\phi}R_{z,\theta}$, where R=

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{pmatrix} \begin{pmatrix} \cos\phi & 0 & -\sin\phi \\ 0 & 1 & 0 \\ \sin\phi & 0 & \cos\phi \end{pmatrix} \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

To represent the rotated point P_{uvw} in terms of the XYZ coordinate system, it is premultiplied by the composite rotation matrix.

$$\begin{pmatrix} P_x \\ P_y \\ P_z \end{pmatrix} = \mathrm{R} \begin{pmatrix} P_u \\ P_v \\ P_w \end{pmatrix}.$$

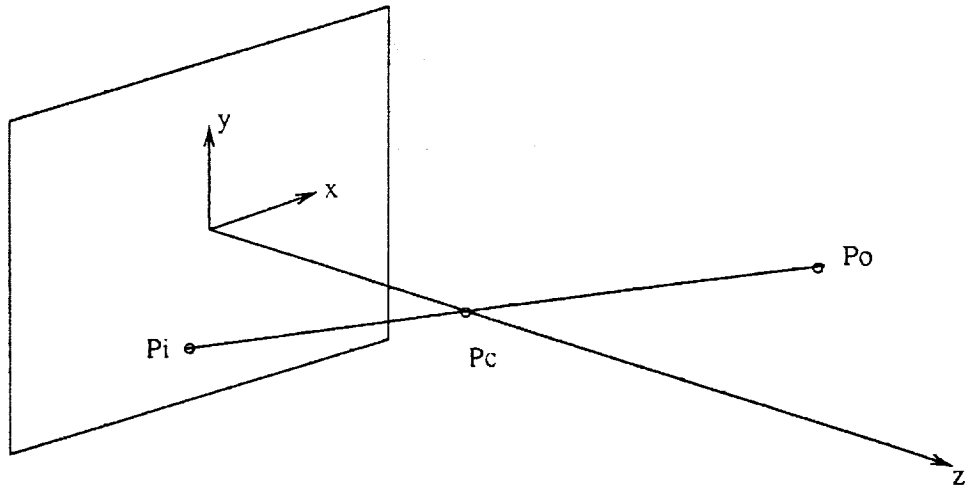

Figure 2.3: Pinhole Camera Model

2.2 Perspective Transformation

Following [13] the perspective projective transformation for a camera is modeled by a pinhole camera. This model maps points from a three-dimensional world space into a two-dimensional image plane. It is initially assumed that the coordinate systems for the image points are coincident and centered in the image plane. This is shown in figure 2.3.

From the above, it is readily observed that any imaged point lies on the plane connecting the object point to the center of the projection. Using this observation, a relationship between the imaged point and the object point is obtained:

$$k(\mathrm{P}_i - \mathrm{P}_c) = (\mathrm{P}_c - \mathrm{P}_o), \tag{2.5}$$

$$k\begin{pmatrix} x_i \\ y_i \\ 0 \end{pmatrix} - \begin{pmatrix} 0 \\ 0 \\ f \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ f \end{pmatrix} - \begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix}, \tag{2.6}$$

$$\begin{pmatrix} kx_i \\ ky_i \\ -kf \end{pmatrix} = \begin{pmatrix} -x_0 \\ -y_0 \\ f - z_0 \end{pmatrix}.$$

Solving for image points in terms of object points, the following is obtained

$$x_i = \frac{-x_0}{k},$$

$$y_i = \frac{-y_0}{k}.$$

Solving for k, it is found that

$$k = \frac{f - z_0}{-f}.$$

finally, solving for the image points in terms of object points and focal length, the following is obtained:

$$x_i = \frac{fx_0}{f - z_0},$$

$$y_i = \frac{fy_0}{f - z_0}.$$

2.3 Homogenous Coordinates

The use of homogeneous coordinates is an extremely useful tool for dealing with coordinate transformations. They provide an efficient matrtix form for the representation of a combination of perspective transformations, rotations, about the x, y, or z axis, scale changes, and translations. As a result the use of homogeneous coordinate extends to the field of computer graphics, robotics, and computer vision. What follows is a brief introduction to the topic.

Homogeneous coordinates essentially transform a n×1 vector into a (n+1) ×1 vector. This is accomplished by multiplying each of the n elements of the origanal vector by a constant scale factor, denoted by w. These scaled quantities become the

first n elements of the new vector. The (n+1) position of the new vector is occupied by the scale factor. This concept can be clarified using the following example. Given a point in three dimensional cartesian space, denoted by the vector,

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix},$$

The homogeneous representation of this point would be denoted by

$$\begin{pmatrix} X/w \\ Y/w \\ Z/w \\ w \end{pmatrix}.$$

It is readily observed that a transformation from homogeneous coordinates back to the original vector space is accomplished by dividing the first n elements of the homogeneous coordinate vector. An example might prove useful in clarifying the transformation. Given a homogeneous coordinate vector

$$\begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix},$$

The cartesian vector is represented as

$$\begin{pmatrix} \frac{a}{d} \\ \frac{b}{d} \\ \frac{c}{d} \end{pmatrix}.$$

The above concept will prove to be extremely useful in the analysis of coordinate transformations.

2.4 Coordinate Transformations Using Homogeneous Coordinates

A homogeneous transformation matrix is defined as an n×n matrix that maps an n dimensional homogeneous vector and transforms it into another homogeneous vector. In the case of a 4×4 transformation matrix, a 4×1 homogeneous vector in one coordinate system is mapped into a 4×1 homogeneous vector in another coordinate system.

For the special case of three-dimensional vector manipulations, the homogeneous transformation matrix can be subdivided into four distinct operations: rotation, translation, scaling, and perspective transformation. Combinations of these transformations can be obtained by multiplying the matrices of the component transformations.

The rotational transformation can be represented as

$$\mathrm{R} = \begin{pmatrix} R_{11} & R_{12} & R_{13} & 0 \\ R_{21} & R_{22} & R_{23} & 0 \\ R_{31} & R_{32} & R_{33} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix},$$

where R is a three dimensional composite rotation matrix.

The translation transformation is defined as

$$\begin{pmatrix} 1 & 0 & 0 & P_x \\ 0 & 1 & 0 & P_y \\ 0 & 0 & 1 & P_z \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

A scale transformation is represented as

$$\begin{pmatrix} k_x & 0 & 0 & 0 \\ 0 & k_y & 0 & 0 \\ 0 & 0 & k_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix},$$

where k_x, k_y, and k_z are scalar factors in the x, y, and z directions respectively.

For the case of a perspective transformation using a pin hole camera model and back projection, the homogeneous transformation matrix can be represented in two forms.

$$\begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & f & 0 \\ -1 & 0 & 0 & f \end{pmatrix} \quad or \quad \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -\frac{1}{f} & 0 & 0 & 1 \end{pmatrix},$$

typically, the latter form is employed.

2.5 Geometric Moments

Moments have been utilized in a wide variety of applications ranging from aircraft to character identification. They are relatively simple to compute and can be made invariant to rotation, scale, and translation. They are one of a general class of shape descriptors. In the presentation that follows the two-dimensional moments are analyzed.

Given a piecewise continuous irradiance function, denoted by f(x,y), the (p+q) ordered moments are

$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q f(x,y)\, dx\, dy, \tag{2.7}$$

$p, q = 0, 1, 2, \ldots$.

It should be noted that the moment sequence m_{pq} is uniquely determined by the irradiance function f(x,y) given that f(x,y) has nonzero values in a finite portion of the plane. As a consequence, the function f(x,y) is uniquely determined by its moment sequence m_{pq}. To have utility in pattern recognition, moments should be invariant to parallel translations, rotations in the plane normal to the optical axis, and scaling.

2.6 Central Moments

It is possible to make the geometric moments invariant to parallel translation. This property is obtained by transforming the geometric moments into central moments. These central moments are defined as

$$\mu_{pq} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(x-\bar{x})^p(y-\bar{y})^q f(x,y)d(x-\bar{x})d(y-\bar{y}), \tag{2.8}$$

where

$$\bar{x} = m_{10}/m_{00}.\ \bar{y} = m_{01}/m_{00}. \tag{2.9}$$

The translation invariance of the central moments is easily shown consider the mapping

$$x\prime = x + h \tag{2.10}$$

$$y\prime = y + k \tag{2.11}$$

which transforms the nonzero region A of f(x,y), into $A\prime$. The central moments of $A\prime$ are

$$\mu\prime_{pq} = \int\int_{A\prime}(x\prime-\bar{x}\prime)(y\prime-\bar{y}\prime)f(x,y)d(x\prime-\bar{x}\prime)d(y\prime-\bar{y}\prime), \tag{2.12}$$

where

$$\bar{x}\prime = m\prime_{10}/m\prime_{00}.\ y\prime = m\prime_{01}/m\prime_{00}. \tag{2.13}$$

Since

$$m\prime_{01} = m_{01} + k, \tag{2.14}$$

$$m\prime_{10} = m_{10} + h, \tag{2.15}$$

and

$$M\prime_{00} = M_{00}, \tag{2.16}$$

substitution into 2.12 yields 2.8.

The central moments can be represented in terms of the ordinary moments.

$$\mu_{pq} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(x-\bar{x})^p(y-\bar{y})^q f(x,y)dxdy \tag{2.17}$$

$$\mu_{pq} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}[\sum_{i=0}^{p}\begin{pmatrix} p \\ i \end{pmatrix} x^i(-\bar{x})^{p-i}][\sum_{j=0}^{q}(j^q y^j(-\bar{y})^{q-j}]f(x,y)dxdy \qquad (2.18)$$

Combining summations

$$\mu_{pq} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}[\sum_{i=0}^{p}\sum_{j=0}^{p}\begin{pmatrix} p \\ i \end{pmatrix}\begin{pmatrix} q \\ j \end{pmatrix}(-\bar{x})^{p-i}(-\bar{y})^{q-j}x^i y^j]f(x,y)dxdy \qquad (2.19)$$

Interchanging summations and integrals

$$\mu_{pq} = \sum_{i=0}^{p}\sum_{j=0}^{q}\begin{pmatrix} p \\ i \end{pmatrix}\begin{pmatrix} q \\ j \end{pmatrix}(-\bar{x})^{p-i}(-\bar{y})^{q-j}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}x^i y^j f(x,y)dxdy \qquad (2.20)$$

From equation 2.7 and equation 2.20 it is clear that

$$\mu_{pq} = \sum_{i=0}^{p}\sum_{j=0}^{q}\begin{pmatrix} p \\ i \end{pmatrix}\begin{pmatrix} q \\ j \end{pmatrix}(-\bar{x})^{p-i}(-\bar{y})^{q-j}M_{ij}. \qquad (2.21)$$

2.7 Algebraic Invariants

Algebraic invariants have surfaced in the works of Lagrange and were rediscovered in the works of Gauss, but neither of these men decided to develop their observations into a formal theory. It was not until Boole, Cayley, and Sylvester that the study of the theory of algebraic invariants flourished.

Hu is usually credited with the application of algebraic invariant theory to the formulation of rotation and scale invariant functions of moments. The derivation that follows is credited to him [1].

Given a binary algebraic form of u and v expressed as

$$f = \sum_{i=0}^{p}\begin{pmatrix} p \\ p-i \end{pmatrix}(a_{(p-i),i})u^{p-i}v^i. \qquad (2.22)$$

or using the Cayley notation as

$$f = (a_{p0}: a_{p-1,1}: \ldots : a_{1,p-1}: a_{0p})(u,v)^p. \qquad (2.23)$$

A homogeneous polynomial of the "a" coefficients is an algebraic invariant of weight w, if

$$I(a\prime_{p0}, \ldots, a\prime_{0p}) = \Delta^w I(a_{p0}, \ldots, a_{0p}) \tag{2.24}$$

where $a\prime_{p0}, \ldots, a\prime_{0p}$ are the coefficients obtained from substituting the following general affine transformation into the original algebraic binary form.

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \alpha & \gamma \\ \beta & \delta \end{bmatrix} \begin{bmatrix} u\prime \\ v\prime \end{bmatrix}. \tag{2.25}$$

and Δ is the determinant of the linear transformation

$$\Delta = \alpha\delta - \beta\gamma \neq 0. \tag{2.26}$$

If the weight of the invariant is zero, it is an absolute invariant; otherwise it is a relative invariant. There exist certain affine transformations that allow Δ to be something other than the determinant of the transformation. These transformations are useful in deriving the necessary moment invariants. It is also useful to introduce another pair of variables, x and y, and subject them to the transformation

$$\begin{bmatrix} x\prime \\ y\prime \end{bmatrix} = \begin{bmatrix} \alpha & \beta \\ \gamma & \delta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}. \tag{2.27}$$

Transformation 2.25 is referred to as the contragredient transformation, and transformation 2.27 is referred to as the cogredient transformation. The eight variables x, y, u, v, $x\prime$, $y\prime$, $u\prime$, and $v\prime$ share the invariant relationship

$$ux + vy = u\prime x\prime = v\prime y\prime. \tag{2.28}$$

To apply the theory of algebraic invariants to moments, it is necessary to define an algebraic binary form which has as its coefficients the moments of order p. One such function is the moment generating function which is defined as

$$M_{u,v} = \int_{-\infty}^{\infty} \sum_{p=0}^{\infty} \frac{1}{p!}(ux + vy)^p f(x,y)dxdy. \tag{2.29}$$

Interchanging integration and summation produces

$$M_{u,v} = \sum_{p=0}^{\infty} \frac{1}{p!} \cdot \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} [\sum_{i=0}^{p} \begin{pmatrix} p \\ p-i \end{pmatrix} x^{p-i} y^i u^{p-i} v^i] f(x,y) dx dy. \tag{2.30}$$

Equation 2.30 is equivalent to

$$\sum_{p=0}^{\infty} \frac{1}{p!} (u_{p0}, \ldots . u_{0p})(u,v)^p. \tag{2.31}$$

By combining equations 2.25, 2.27, and 2.29 the following is obtained:

$$M\prime(u\prime, v\prime) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \sum_{p=0}^{\infty} \frac{1}{p!} (u\prime x\prime + v\prime y\prime)^p f\prime(x\prime, y\prime) \frac{1}{|J|} dx\prime dy\prime \tag{2.32}$$

where J is the jacobian of transformation 2.27, $f\prime(x\prime, y\prime)$ is equal to f(x,y) and $M\prime(u\prime,v\prime)$ is the moment generating function of the transformation. Since

$$\mu\prime_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x\prime)^p (y\prime)^q f(x\prime, y\prime) dx\prime dy\prime, \tag{2.33}$$

$$M\prime(u\prime, v\prime) = \frac{1}{|J|} \sum_{p=0}^{\infty} \frac{1}{p!} (u\prime_{p0}, \cdots, u\prime_{0p})(u\prime, v\prime)^p. \tag{2.34}$$

Combining the results of 2.29, 2.30, 2.31, 2.32 and 2.34 it is shown that if the binary algebraic form of order p has an algebraic invariant then the *pth* order moments have the same invariant but multiplied by the absolute value of the Jacobian of the cogredient transformation. In other words if

$$I(a\prime_{p0}, \ldots . a\prime_{0p}) = \Delta^w I(a_{p0}, \ldots, a_{0p}), \tag{2.35}$$

then

$$I(u\prime_{p0}, \ldots . u\prime_{0p}) = |J|^{\Delta w} I(u_{p0}, \ldots, u_{0p}). \tag{2.36}$$

Under the scale change denoted by

$$\begin{bmatrix} x\prime \\ y\prime \end{bmatrix} = \begin{bmatrix} \alpha & 0 \\ 0 & \alpha \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}, \tag{2.37}$$

each coefficient of the binary algebraic form is an invariant

$$a'_{pq} = \alpha^{p+q} a_{pq}. \tag{2.38}$$

Therefore, the relative moment invariants are multiplied by the Jacobian of 2.37 producing

$$\mu'_{pq} = \alpha^{p+q+2} \mu_{pq}. \tag{2.39}$$

To obtain an absolute scale invariant, the value of α is obtained from the relationship between the zeroth order moments.

$$\alpha = \frac{\mu'_{00}}{\mu_{00}}. \tag{2.40}$$

substituting 2.40 into 2.39 obtains:

$$\frac{\mu'_{pq}}{(\mu'_{00})^{(\frac{p+q}{2}+1)}} = \frac{\mu_{pq}}{(\mu_{00})^{(\frac{p+q}{2}+1)}}. \tag{2.41}$$

What follows is a derivation of rotational invariance. For a rotational transformation the contragredient transformation is

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} u' \\ v' \end{bmatrix}. \tag{2.42}$$

and the cogredient transformation is

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}. \tag{2.43}$$

Since the Jacobian of the cogredient transformation is equal to one, the algebraic invariant is equivalent to the moment invariant. Therefore, treating the moments as the coefficient of the binary form

$$(\mu_{p0}, \ldots, \mu_{0p})(u, v)^p, \tag{2.44}$$

and using the following transformation

$$\begin{bmatrix} t \\ v \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & i \\ 1 & -i \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} \tag{2.45}$$

and

$$\begin{bmatrix} U' \\ V' \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 1 & i \\ 1 & -i \end{bmatrix}\begin{bmatrix} u' \\ v' \end{bmatrix}, \tag{2.46}$$

the following relations are obtained:

$$U' = Ue^{-i\theta}, V' = Ve^{i\theta} \tag{2.47}$$

Substituting 2.47, 2.46, and 2.45 into 2.44,

$$(I_{p0}, \ldots, I_{0p})(U, V)^p \equiv (\mu_{p0}, \ldots, \mu_{0p})(u, v) \tag{2.48}$$

$$(I_{p0}, \ldots, I_{0p})(U, V)^p \equiv (\mu'_{p0}, \ldots, \mu'_{0p})(u', v')^p \tag{2.49}$$

$$(I_{p0}, \ldots, I_{0p})(U, V)^p \equiv (I'_{p0}, \ldots, I'_{0p})(Ue^{-i\theta}, Ve^{-i\theta})^p \tag{2.50}$$

equating like terms in 2.49 obtains

$$I'_{p0} = e^{ip\cdot\theta} I_{p0} \tag{2.51}$$

$$I'_{p-1,1} = e^{i(p-2)\theta} I_{p-1,1}; \ldots; \tag{2.52}$$

$$I'_{1,p-1} = e^{-i(p-2)\theta} I_{1,p-1}; \tag{2.53}$$

$$I'_{0p} = e^{-ip\theta} I_{0p}. \tag{2.54}$$

From the identity of the first two expressions, it is clear that $I_{p-r,r}$ is the complex conjugate of $I_{r,p-r}$ and

$$I_{p-r,r} =$$

$$[(\mu_{p0}; \mu_{p-2,2}; \ldots; \mu_{p-2r,2r})(1,1)^r; (\mu_{p-1,1}; \mu_{p-3,3}; \ldots; \mu_{p-2r-1,2r+1})(1,1)^r; \ldots;$$

$$(\mu_{2r,p-2r}; \mu_{2r+2,p-2r-2}; \ldots; \mu_{0p})(1,1)^r](1,-i)^{p-2r},$$

where p - 2r > 0, and

$$I_{p/2,p/2} = \mu_{p0} + \begin{pmatrix} p/2 \\ 1 \end{pmatrix} + \begin{pmatrix} p/2 \\ 2 \end{pmatrix} \mu_{p-4,4} +, \ldots, +\mu_{0p}, \tag{2.55}$$

where p is even.

For rotation and reflection the cogredient transformation is

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ \sin\theta & -\cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}. \tag{2.56}$$

and

$$I'_{p0} = e^{-ip\theta} I_{0p} \tag{2.57}$$

$$I'_{p-1,1} = e^{-i(p-2)\theta} I_{1,p-1}; \cdots; \tag{2.58}$$

$$I'_{1,p-1} = e^{i(p-2)\theta} I_{p-1,1}; \tag{2.59}$$

$$I'_{0p} = e^{ip\theta} I_{p0}. \tag{2.60}$$

From the above derivation Hu obtains six rotation invariants and one skew invariant. They are as follows:

$$\mu_{20} + \mu_{02}$$

$$(\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2$$

$$(\mu_{30} - 3\mu_{12})^2 + (3\mu_{21} - \mu_{03})^2$$

$$(\mu_{30} + \mu_{12})^2 + (\mu_{21} + \mu_{03})^2$$

$$(\mu_{30} - 3\mu_{12})(\mu_{30} + \mu_{12})[(\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2] +$$
$$(3\mu_{21} - \mu_{03})(\mu_{21} + \mu_{03})[3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2)]$$

$$(\mu_{20} - \mu_{02})[(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2] + 4\mu_{11}[(\mu_{30} + \mu_{12})(\mu_{21} + \mu_{03})]$$

$$(3\mu_{21} - \mu_{03})(\mu_{30} + \mu_{12})[(\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2] -$$
$$(\mu_{30} - 3\mu_{12})(\mu_{21} + \mu_{03})[3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2].$$

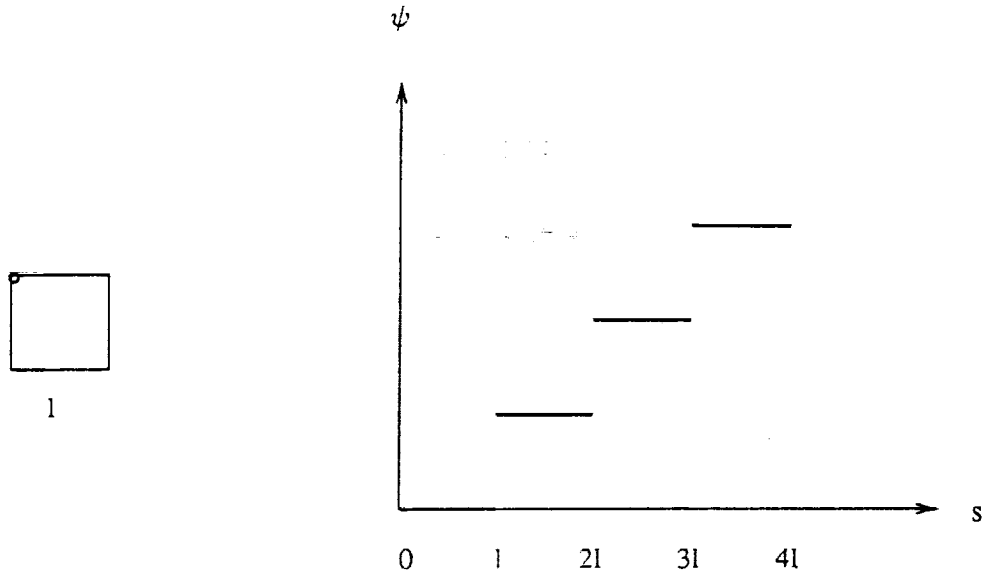

Figure 2.4: Square and corresponding ψ - s curve

2.8 ψ - s curves

The utilization of ψ - s curves is one way to characterize the shape of an image using its boundary. It is essentially a chain coded representation of the boundary. ψ is the angle formed between the a reference line and the tangent to the curve, and s is the arc length as the boundary is traversed. It can be shown that staight lines in an image correspond to horizontal lines in the ψ - s curve and circles correspond to straight lines with slopes of $\frac{1}{r}$. The ψ - s curve for a closed boundary is periodic with a discontinuous jump from 2π to 0 as the curve is retraced to the starting point. Figures 2.4, 2.5, and 2.6 show the ψ - s curves for a square, circle, and equilateral triangle, respectively.

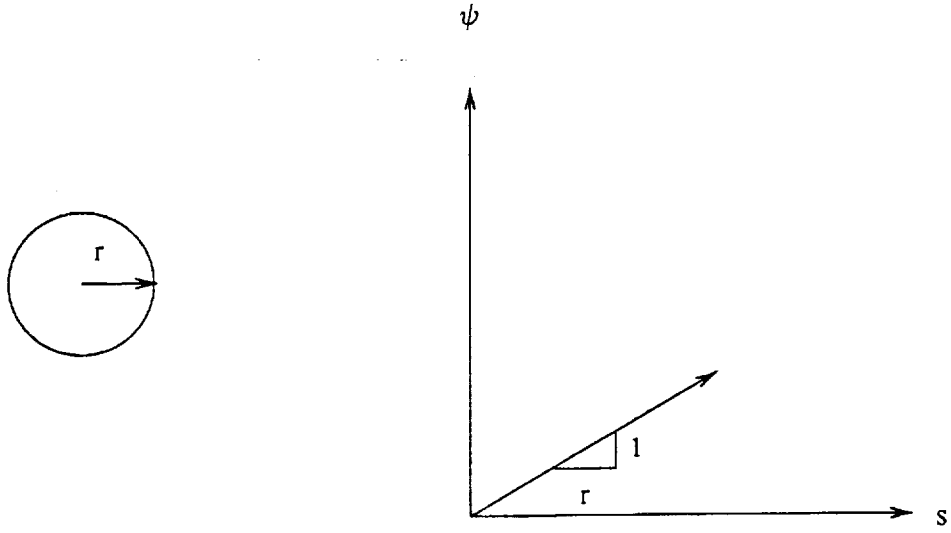

Figure 2.5: Circle and corresponding ψ - s curve

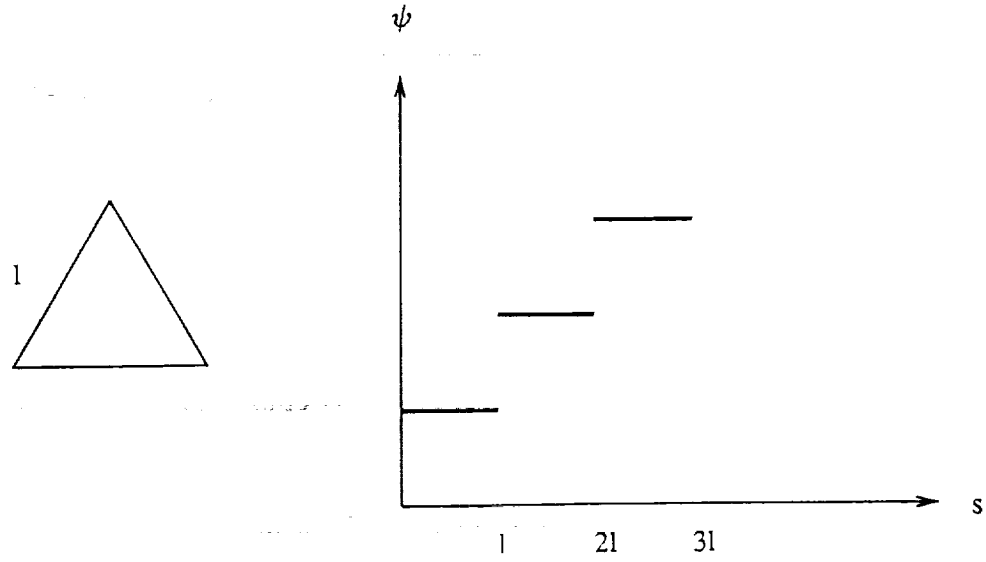

Figure 2.6: Equilateral triangle and corresponding ψ - s curve

CHAPTER 3
PROBLEM STATEMENT

The objective consists of determining the position of each link of a PUMA arm using fiducial marks. Each link should be uniquely identifiable regardless of the orientation of the arm providing that the link is in the field of view of the camera. The placement of multiple fiducial marks on the arm provides an excellent method to accomplish this task. The methodology behind utilizing multiple fiducial marks to label each link of the PUMA is task efficient and effective. The fiducial mark itself is a planar object of specified dimensions. Because of the dimensional specification, the location of each affixed mark is known relative to the arm-centered coordinate system. This reduces the original objective to one of distinquishing fiducial marks and locating an associated point.

When the marks are viewed by the CCD camera all the points on the fiducial marks are subjected to a perspective projective transformation that maps the three dimensional coordinates of the mark into two dimensional points in the image plane. This mapping is a noninvertible transformation. Therefore, any point in the image plane can correspond to an infinite number of points, in the arm-centered coordinate system that lie upon the line connecting the image point to the focus of the camera. However, it is possible to locate the arm-centered coordinates of the mark by utilizing two calibrated cameras. If an algorithm is employed to locate a particular point in both image planes the two locations can be used in a triangulation algorithm to identify the location of the point in the arm-centered coordinate system. The triangulation algorithm is straightforward. Therefore, it will not be addressed any further.

If the arm-centered coordinate system is aligned with the image coordinate system, the perspective transformation would take the form given in Chapter Two.

For the two coordinate systems to be aligned, it is necessary to align the x-axis and y-axis of the arm-centered system with the u-axis and v-axis of the image, respectively. Since this is generally not the case, it is necessary to pre-multiply the arm-centered coordinates by a composite rotation and translation matrix that transforms the original coordinates to coordinates relative to a three-dimensional coordinate system that has its origin aligned with the image plane origin, its x-axis aligned with the u-axis, its y-axis aligned with the v-axis, and its z-axis aligned with the optical axis. Once the linear transformation is achieved the perspective transformation given in Chapter Two is valid. Since the cameras are calibrated and the transformation is known, obtaining the coordinates of the point in the arm-centered coordinate system is accomplished by post-multplying the coordinates obtained via triangulation with the inverse of the transformation matrix. Therefore, it is evident that the recognition of the system of marks and their associated points determines the position and orientation of the arm. The main emphasis of this work is the design and recognition of the fiducial marks.

The imaging of an object using a CCD camera produces a substantial amount of distortion. Perspective transformation, quantization and sampling produce the most distortion, but the pincushion and barrel effect also contribute to the degradation of the object representation. As a consequence, it is necessary to design marks and feature extraction algorithms that are insensitive to these effects. Perspective distortion of a mark occurs when there are points in the mark that have different optical axis coordinate values. It is essentially the converging railroad effect. It can transform squares into trapezoids and circles into distorted ellipses. Because it is proportional to the inverse of the distance along the optical axis, it is difficult to account for without an approximate knowledge of the position and orientation of the mark. When all the points within a mark lie in a plane perpendicular to the optical axis only a scale change results. Therefore, if successive images are taken

of a mark that is only translated in the direction parallel to the optical axis, the images only differ by a scale factor.

Representation of a continuous object by a finite number of pixels inherently produces an inaccurate representation. As the ratio of the image size to pixel size decreases, the image distortion increases. If the ratio of image size to pixel size becomes too small, the image becomes unrecognizable. This effect is similiar to the aliasing effect for one-dimensional periodic signals.

For the particular robotic system employed in this work, the mark will be no greater than two meters away from the camera. The resolution of the frame grabber is 512×480 pixels and the field of view is approximately two square meters. This produces a pixel resolution of approximately two millimeters when the object is two meters away from the image plane. This implies that the marks should be as large as possible to compensate for the large pixel size at that distance, but there is a limitation on the size of the pixels. This limitation is caused by the link size. Each link has six sides and at least four of these sides can be used to affix a mark. The smallest side of a link is approximately 3.5 inches. Therefore, this is the upper bound of the size of the fiducial mark.

CHAPTER 4
FIDUCIAL MARK IDENTIFICATION

The problem consists of generating a sufficient number of fiducial marks to label the robot arm reqardless of its position and orientation in space and the background it is placed upon.

4.1 Design of the Fiducial Marks

It was decided that marks based upon simple geometric figures might aid in the identification process. In any recognition process involving several patterns it is necessary to extract a set of features that when utilized in a decision making function will yield a unique value for each of the patterns. If this criterion is not satisfied then two or more of the patterns cannot be distinquished. For this reason the circle, square, and equilateral triangle seemed like excellent candidates for fiducial marks. They possess features readily extracted and uniquely determined. Some of the features that can be extracted are moment invariants, contour signatures, and compactness measures.

Another criterion that needs to be addressed is the size restriction of the marks. The marks are restricted to a 3.5 inch box. This restriction exists due to the fact the marks have to be placed on the arm. Each mark must fit on each face of every link.

Another criterion that needs to be addressed is the quantity of unique marks used to label the links of the arm. The three basic shapes must generate at least twenty-seven unique composite shapes to accomplish the labeling task. This can be accomplished by nesting shapes within other shapes. The sizes of the shapes are chosen to maximize the size of the inner shape while insuring that there is at least a three pixel wide border separating the shapes. The inner shape size is maximized

to minimize the distortion due to digitizing. As the ratio of the shape size to pixel size is decreased the shapes become increasingly more difficult to distinquish. All the shapes designed consist of a black shape within a white shape within a black shape within a white rectangular border. This configuration generates twenty-seven unique composite figures.

4.2 Segmentation of the Marks from the Background

For the algorithm to succeed, it is necessary to segment the mark from its surroundings. The placement of the shapes within a white rectangular region enables the composite patterns to remain intact. If the outer white rectangular region is not present and the marks are placed upon a black background, the outer figure might be unrecoverable in the image. It is possible for the outer white rectangular region to be distorted by its background: but this is of no consequence because the algorithm only searches for a white border and doesn't try to classify the shape. This algorithm is extremely efficient for backgrounds with a relatively small number of white regions. After a white region is located, the algorithm searches the inner region to determine the presence of a mark.

4.3 Extracting the Outer, Inner, and Middle Borders

Extraction of each of the borders is vitally important in obtaining a reliable feature space. If an error is produced in the border extraction process, the subsequent feature space calculations will yield inaccurate results. In general, the method chosen to extract the border depends upon the border definition. For a continuous image a boundary point is usually defined as follows

Definition 1 *A boundary point of a set is a point having the property that every neighborhood of it contains points in the set and points not in the set.*

This definition usually refers to the set of points within some connected region and the set of points outside the region, where each region consists of an infinite number of points, and the neighborhood of each point is infinite. For a discrete image every region contains an finite set of points or pixels and each neighborhood consists of at most eight pixels. Because of this distinction, it is necessary to modify the above definition. Before a definition of the border pixel for a discrete image is given, it is necessary to define two related terms. These terms are the four-neighbors and the eight-neighbors of a pixel.

Definition 2 *Given a pixel P at coordinates (x,y), the four-neighbors of the pixel are given by the pixels with the coordinates (x-1,y), (x,y+1), (x+1,y), and (x,y-1).*

Definition 3 *Given a pixel P at coordinates (x,y), the eight-neighbors of the pixel are given by the four-neighbors of the pixel and the pixels with the additional coordinates (x-1,y+1), (x+1,y+1), (x+1,y-1), and (x-1,y-1).*

Now, the definition of a border pixel can proceed.

Definition 4 *A pixel P at coordinates (x,y) is a border pixel if and only if P has at least two eight-neighbors in the same set as P, and P does not have more than three four-neighbors in the same set as P, where the set P contains all of the the pixels that have intensities that are allowed to be connected.*

For binary images the images are divided into two sets, pixels with a value of one and pixels with a value of zero.

To extract the boundary of each shape, the algorithm searches for a boundary candidate. Once a candidate is found it searches the eight-neighbors of this pixel in a clockwise fashion for another border candidate. If another candidate is found it searches the eight-neighbors of this pixel. It continues this search until it reaches the first border element. If at any point it cannot find another border pixel, it returns

to the previous pixel and searches the remaining eight-neighbors. If the algorithm backtracks to the starting point and cannot find any border elements among the remaining eight-neighbors it returns and moves on to the other starting points. If no borders are found in the admissable region a failed flag is returned.

Once the border from an enclosing shape is obtained, the algorithm restricts its search region to the region enclosed by the border and searches for border elements that belong to the opposite intensity set of the enclosing border.

4.4 Moments as Feature Parameters

Moments and functions of moments have been utilized as pattern features in a number of applications involving the recognition of planar objects. Functions of moments can be utilized to obtain features which are invariant to scale, rotation and translation. They are considered to be reliable features if they are insensitive to image degrading effects such as quantization, and sampling. Moments are global descriptors which characterize the distribution of the points of an image. One of the major drawbacks of using moments is the large number of multiplications involved in the computational process. The straightforward method of calculating moments requires 10MN multiplications for an $M \times N$ image. Unless the system has a dedicated math coprocessor, the extraction of these features in real time is infeasible. However, there have been several fast algorithms devised for this problem.

4.4.1 Moments of a Generalized Rectangle

Consider the generalized rectangle represented by Figure 4.1 where a_1 is the width, $T(a_1)$ is the length, and T is the arbitrary scale factor. Consider the region in the u-v plane represented by figure 4.2

The central moments of the region are denoted as follows:

$$\mu_{pq} = \int_{-a_1}^{a_1} \int_{-T a_1}^{T a_1} x^p y^q dx dy \tag{4.1}$$

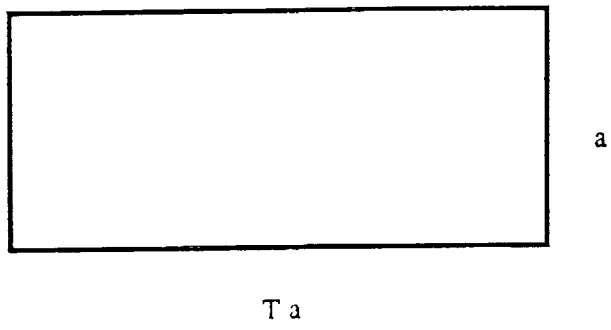

Figure 4.1: Generalized Rectangle

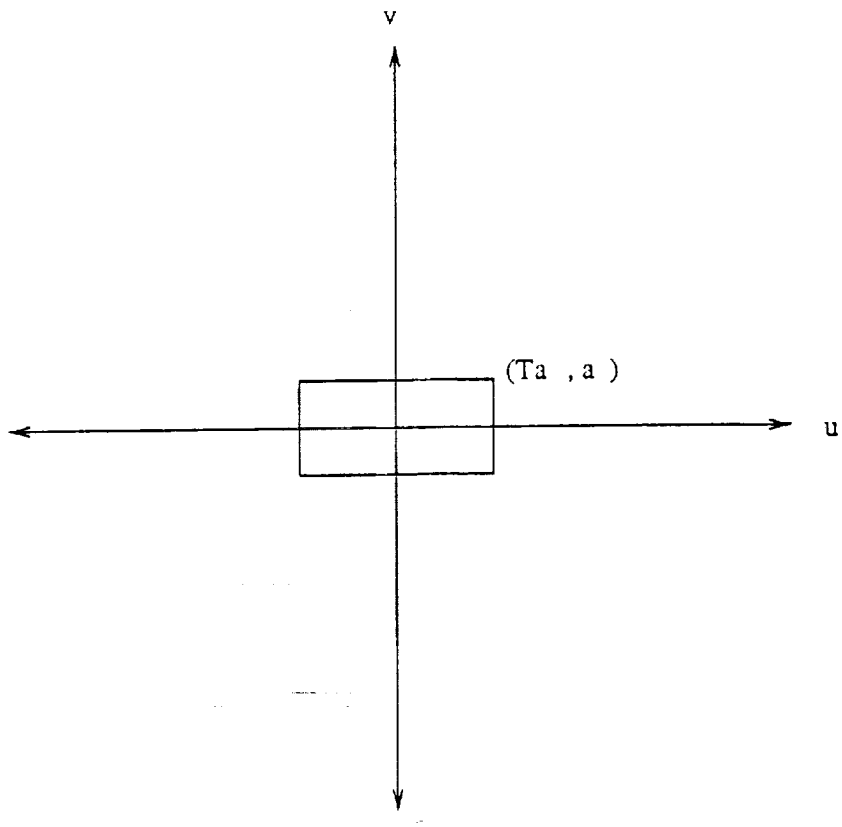

Figure 4.2: Rectangular region in the u-v plane

$$\mu_{pq} = \int_{-a_1}^{a_1} [\frac{x^{p+1}}{p+1}|_{-Ta_1}^{Ta_1}] y^q dy \tag{4.2}$$

$$\mu_{pq} = [\frac{x^{p+1}}{p+1}|_{-Ta_1}^{Ta_1}][\frac{y^{q+1}}{q+1}|_{-a_1}^{a_1}] \tag{4.3}$$

$$\mu_{pq} = [\frac{(Ta_1)^{p+1}}{p+1} - \frac{(-Ta_1)^{p+1}}{p+1}][\frac{a_1^{q+1}}{q+1} - \frac{-a_1^{q+1}}{q+1}] \tag{4.4}$$

$$\mu_{pq} = \frac{1}{(p+1)(q+1)}[(Ta_1)^{p+1}(1-(-1)^{p+1})][a_1^{q+1}(1-(-1)^{q+1})] \tag{4.5}$$

$$\mu_{pq} = \frac{1}{(p+1)(q+1)}[T^{(p+1)}a_1^{(p+q+2)}][(1-(-1)^{p+1})(1-(-1)^{q+1})] \tag{4.6}$$

It can be readily seen that if either p or q is odd then $\mu_{pq} = 0$. If p and q is even then μ_{pq} is given by

$$\mu_{pq} = \frac{1}{(p+1)(q+1)}[4T^{(p+1)}a_1^{(p+q+2)}]. \tag{4.7}$$

To calculate N_{pq}, the scale normalized central moments, the following is used

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\frac{p+q}{2}+1}}. \tag{4.8}$$

Using the results of equation 4.8

$$\eta_{pq} = \frac{4T^{(p+1)}a_1^{(p+q+2)}}{(4Ta_1^2)^{(\frac{p+q}{2}+1)}(p+1)(q+1)} \tag{4.9}$$

$$\eta_{pq} = \frac{4T^{p+1}a_1^{(p+q+2)}}{4^{(\frac{p+q}{2}+1)}T^{(\frac{p+q}{2}+1)}a_1^{(p+q+2)}(p+1)(q+1)} \tag{4.10}$$

$$\eta_{pq} = \frac{T^{(\frac{p-q}{2})}}{4^{(\frac{p+q}{2})}(p+1)(q+1)} \tag{4.11}$$

The first rotationally invariant moment, ϕ_1, is given by

$$\phi_1 = \eta_{20} + \eta_{02}, \tag{4.12}$$

where η_{20} is

$$\eta_{20} = \frac{T}{12} \tag{4.13}$$

and η_{02} is

$$\eta_{02} = \frac{1}{12T} \tag{4.14}$$

As a result,

$$\phi_1 = \frac{T + 1/T}{12} \tag{4.15}$$

For the special case of a square, T=1.

$$\phi_1 = 1/6.$$

For the case where T=2.

$$\phi_1 = 5/24$$

the second rotationally invariant moment ϕ_2 is given by

$$\phi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2.$$

Since η_{11}=0.

$$\phi_2 = \frac{(T - 1/T)^2}{12^2} \tag{4.16}$$

It is clear from equations 4.15 and 4.16 that the moment function $\phi_1^2 - \phi_2$ is invariant for all rectangles.

4.4.2 Moment Calculations for a Generalized Ellipse

The calculation of the moments of a generalized ellipse is sufficient to obtain the moments of a circle. It was decided to extract these features. This extraction may prove to be useful in the analysis of the effects of the distortion of a circle introduced by orthogonal projection. If the ellipse is represented by Figure 4.3, the central moments of the region under the change of variables

$$x = Tr_1 \cos\theta$$

and

$$y = r_1 \sin\theta$$

is denoted by

$$\mu_{pq} = \int_0^{2\pi} \int_0^{r_1} (Tr\cos\theta)^p (r\sin\theta)^q Tr dr d\theta.$$

Grouping r's and extracting the scale factor produces

$$\mu_{pq} = (T^{p+1}) \int_0^{2\pi} \int_0^{r} r^{(p+q+1)} \cos^p \sin^q \theta dr d\theta$$

The first iterated integration produces

$$\mu_{pq} = \frac{T^{(p+1)} r_1^{(p+q+2)}}{(p+q+2)} \int_0^{2\pi} \cos^p \sin^q \theta d\theta$$

The evaluation of the second iterated integral produces

$$\mu_{pq} = \frac{T^{(p+1)} r_1^{(p+q+2)}}{(p+q+2)} [\frac{(p-1)(p-3)\cdots 3\cdot 1}{(p+q)(p+q-2)\cdots(p+2)}][\frac{(q-1)\cdots 3\cdot 1}{q\cdots 4\cdot 2}]2\pi \qquad (4.17)$$

for p and q even, and zero otherwise.

Substitution of p=2 and q=0 into equation 4.17 yields

$$\mu_{20} = \frac{T^3 r_1^4}{4}\pi.$$

Substitution of p=0 and q=2 into equation 4.17 yields

$$\mu_{02} = \frac{T r_1^4}{4}\pi$$

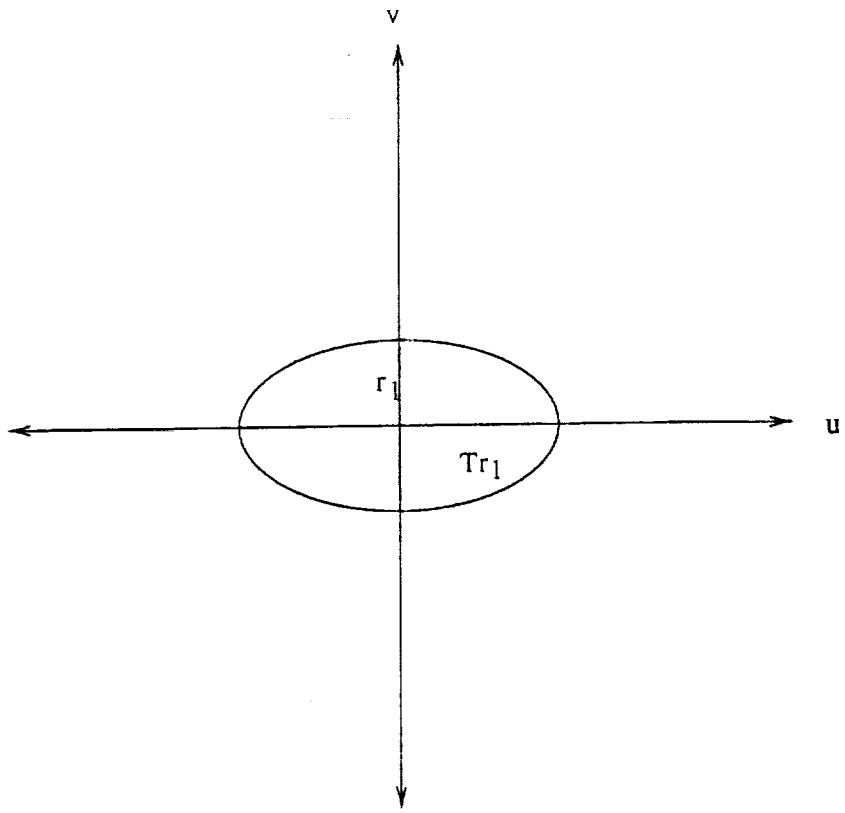

Figure 4.3: Ellipse in the u-v plane

Calculation of the second order scale invariant moments produces

$$\eta_{20} = \frac{T}{4\pi}$$

$$\eta_{02} = \frac{1}{4T\pi}$$

The first and second rotation, scale and translation invariant moments are

$$\phi_1 = \frac{T + 1/T}{4\pi} \tag{4.18}$$

and

$$\phi_2 = [\frac{T - 1/T}{4\pi}]^2. \tag{4.19}$$

It is clear from equations 4.18 and 4.19 that the moment function $\phi_1^2 - \phi_2$ is invariant for all ellipses.

4.4.3 Moment Calculations for an Equilateral Triangle

Given the equilateral triangle in Figure 4.4 with side length 2a, the central moments are defined as follows:

$$\mu_{pq} = \int_{-a/\sqrt{3}}^{2a/\sqrt{3}} \int_{\frac{1}{3^{1/2}}[y-\frac{2a}{3^{1/2}}]}^{\frac{-1}{3^{1/2}}[y-\frac{2a}{3^{1/2}}]} x^p y^q dx dy$$

Performing the first integration yields

$$\mu_{pq} = \int_{\frac{-a}{3^{1/2}}}^{\frac{2a}{3^{1/2}}} \frac{x^{(p+1)}}{(p+1)} \Big|_{\frac{1}{3^{1/2}}[y-\frac{2a}{3^{1/2}}]}^{\frac{-1}{3^{1/2}}[y-\frac{2a}{3^{1/2}}]} y^q dy.$$

Substituting for x.

$$\mu_{pq} = \frac{1}{p+1} \int_{\frac{-a}{3^{1/2}}}^{\frac{2a}{3^{1/2}}} \frac{-1}{3^{1/2}}(y - \frac{2a}{3^{1/2}})^{p+1} y^q dy$$

for p even.

$$\mu_{pq} = \frac{-2}{(p+1)3^{(p+1)/2}} \int_{\frac{-a}{3^{1/2}}}^{\frac{2a}{3^{1/2}}} [y - \frac{2a}{3^{1/2}}]^{p+1} y^q dy.$$

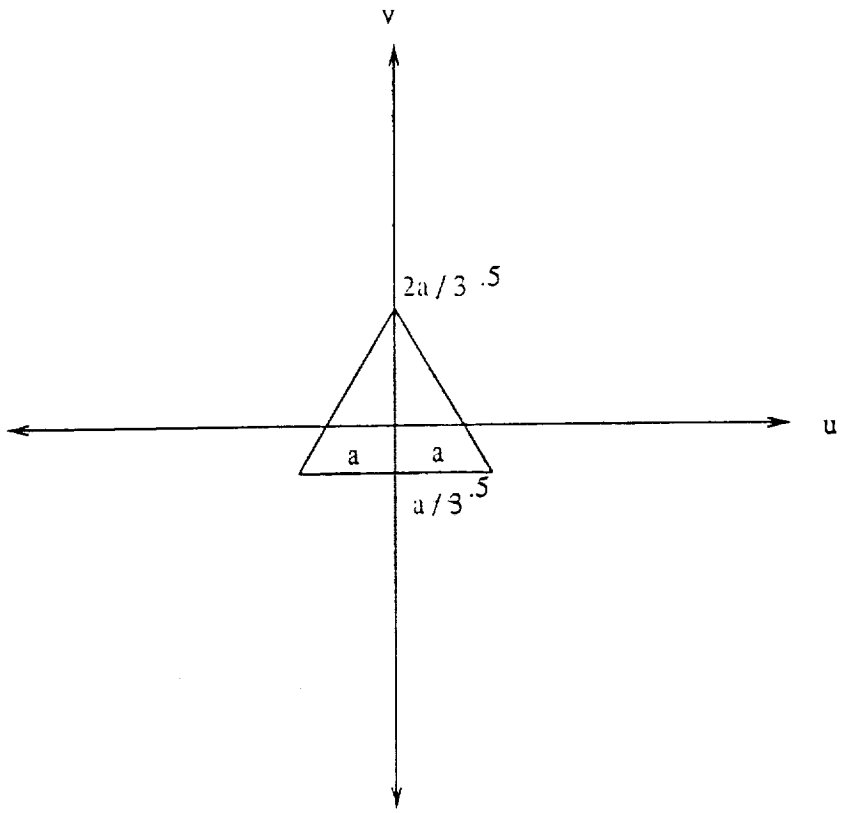

Figure 4.4: Equilateral triangle in the u-v plane

Using the binomial expansion.

$$\mu_{pq} = \frac{-2}{(p+1)3^{(p+1)/2}} \int_{\frac{-a}{3^{1/2}}}^{\frac{2a}{3^{1/2}}} \sum_{i=0}^{p+1} \begin{pmatrix} p+1 \\ i \end{pmatrix} y^{p+1-i} (\frac{-2a}{3^{1/2}})^i y^q dy$$

Performing the second integration yields

$$\mu_{pq} = \frac{-2}{(p+1)3^{(p+1)/2}} \sum_{i=0}^{p+1} \frac{y^{p+q+2-i}}{p+q+2-i} (\frac{-2a}{3^{1/2}})^i |_{\frac{-a}{3^{1/2}}}^{\frac{2a}{3^{1/2}}}$$

substituting for y.

$$\mu_{pq} = \frac{-2}{(p+1)3^{(p+1)/2}} \sum_{i=0}^{p+1} \frac{1}{p+q+2-i} [\frac{-2a}{3^{1/2}}]^i ([\frac{2a}{3^{1/2}}]^{p+q+2-i} - [\frac{-a}{3^{1/2}}]^{p+q+2-i})$$

4.4.4 Moments of orthogonally projected shapes

In many situations, it becomes possible to approximate the perspective projective transformation of an imaging device by an orthogonal transformation. One such situation is the case when the variation in object point distances is negligible with respect to the object plane to image plane distance. In this case the distance of the object along the optical axis may be considered fixed. For example, since

$$x_i = \frac{f x_o}{z_o - f},$$

$$y_i = \frac{f y_o}{z_o - f}$$

and f and z_o are fixed, the transformation between object and image points is equivalent to a scale change.

$$xi = kx_o$$

and

$$y_i = ky_o$$

where k is the constant $\frac{f}{z_o - f}$. This is the orthograpic projection model, and it facilitates the development of moment functions for the basic shapes.

Consider the situation where one of the basic shapes is arbitrarily rotated about the object centered coordinate system. This can be represented by the multiplication of the composite rotation matrix R with each object point $[x'_o\ y'_o\ z'_o]$, where R is given by

$$\begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

and r_{ij} is a trigonometric function of the rotation angles α and ϕ given in Chapter Two. Under this rotation and orthogonal projection,

$$x_i = k(r_{11}x'_o + r_{12}y'_0 + T_x)$$

$$y_i = k(r_{21}x'_0 + r_{22}y'_0 + T_y)$$

Given the endpoints of two parallel line segments in the object coordinate system denoted by for line 1

$$(x_1, y_1)(x_1 + \Delta x, y_1 + \Delta y)$$

and for line 2

$$(x_2, y_2)(x_2 + \Delta x, y_2 + \Delta x).$$

Now if both line segments are subjected to the same rotation and orthogonal projection, the points of line 1 transform into

$$k(r_{11}x_1 + r_{12}y_1 + T_x, r_{21}x_1 + r_{22}y_1 + T_y) \tag{4.20}$$

and

$$k(r_{11}(x_1 + \delta x) + r_{12}(y_1 + \Delta y) + T_x, r_{21}(x_1 + \delta x) + r_{22}(y_1 + \Delta y) + T_y),$$

and those of line 2 transform into

$$k(r_{11}x_2 + r_{12}y_2 + T_x, r_{21}x_2 + r_{22}y_2 + T_y) \tag{4.21}$$

and

$$k(r_{11}(x_2 + \Delta x) + r_{12}(y_2 + \Delta y) + T_x), r_{21}(x_2 + \Delta x) + r_{22}(y_2 + \Delta y) + T_y)$$

From equations 4.20 and 4.21 it is clear that the slopes of both transformed line segments are equivalent and the slope is given by

$$\frac{r_{22}\Delta y + r_{21}\Delta x}{r_{21}\Delta y + r_{11}\Delta x}$$

The line segment lengths are given by

$$\sqrt{(r_{21}\Delta x + r_{22}\Delta y)^2 + (r_{21}\Delta y + r_{11}\Delta x)^2}$$

Since under rotation and orthogonal projection, parallel line segments remain parallel, it is evident that a square under this transformation must be converted to a rectangle. As a result, the moments of an orthogonally projected square are given by those of the generalized rectangle, where

$$T = \frac{\sqrt{r_{11}^2 + r_{21}^2}}{\sqrt{r_{12}^2 + r_{22}^2}} \tag{4.22}$$

Similarily, rotation and orthogonal projection of a circle produces an ellipse and T is given by equation 4.22.

4.4.5 Moment approximations due to digitization

For digital images, the double integration used to define the moments is typically approximated by double summations and the moments are denoted as

$$m_{pq} = \sum_{x=0}^{M}\sum_{y=0}^{N} x^p y^q f(x,y) \tag{4.23}$$

$$\mu_{pq} = \sum_{x=0}^{M}\sum_{y=0}^{N}(x - \bar{x})^p(y - \bar{y})^q f(x,y), \tag{4.24}$$

where M and N are the image dimensions.

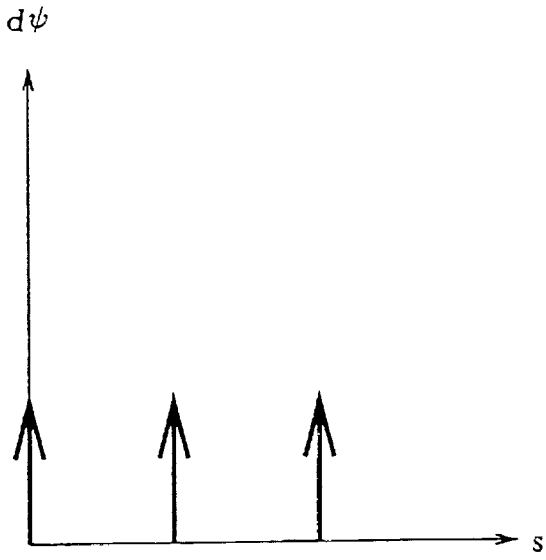

Figure 4.5: Curvature plot of an equilateral triangle

Inherent in this approximation is the loss of strict rotational and scale invariance. For a square it is shown by Teh and Chin [11], that the first invariant moment is

$$\frac{1 - 1/a^2}{6}$$

where a is the ratio of the square size to the pixel size. It is readily observed that ϕ_1 is no longer scale invariant, but depends on the size of the sampling grid. The loss of rotational and scale invariance arises because the sampling grid is not adequate to represent the shapes, and as a result changes in orientation result in a changed representation of the object in the image plane.

4.5 Curvature as a feature parameter

Since most if not all of the shape information of an image is contained in its boundary. The use of features based on the ψ - s curves of an image boundary seems like an exceptable method to obtain unique features. One feature that might yield promising results is the curvature of the boundary.

The curvature of a boundary is defined as the rate of change of ψ with respect

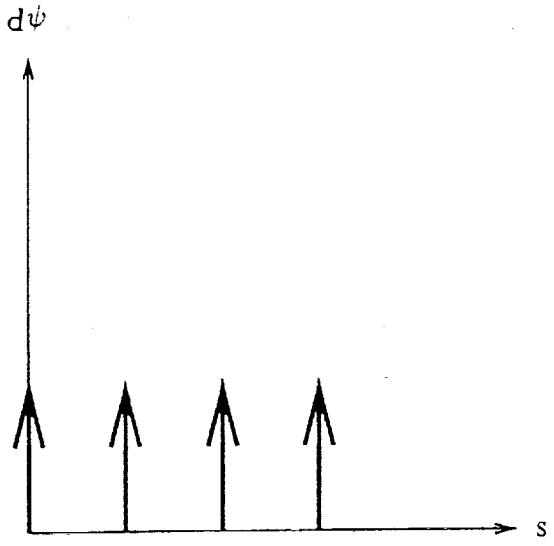

Figure 4.6: Curvature plot of a square

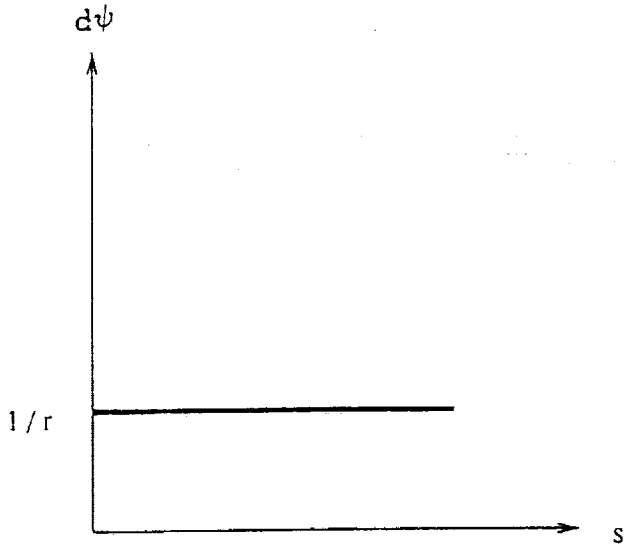

Figure 4.7: Curvature plot of a circle

to the arc length, and it can be easily obtained from the representation of the ψ - s curve of the boundary of an image. Analyzing the curvature plots in Figures 4.5, 4.6, and 4.7 it is readily observed that there are exactly three jump discontinuities for the triangle, four discontinuities for the square, and no discontinuities for the circle, and in general any quadrilateral will have four discontinuities, any triangle will have three discontinuities, and any ellipse will have no discontinuities. This information can be used to discriminate between the three shapes even if they are perspectively distorted. If the boundary of an image is traced and the jumps in curvature are counted, the shape will be determined. This formulation is based on the assumption of a continuous image. Since the image is not continuous, a modification has to be devised.

For a discrete image, the typical ψ - s curve can be represented as an eight directional chain code of the angles. This decreases the feasible angle space of an image from infinity (in the continuous case) to eight, and Δs will either be 1 or $\sqrt{2}$. As a result of the sampling, straight lines in the actual object will not correspond to straight lines in the chain coded version of the image boundary. The chain code essentially links angles that form an approximation of the slope of the line if the angles are averaged. If n is the number of link angles averaged, the approximation will be within

$$\pm \arctan(1/n)$$

of the actual slope of the line. Using the difference of the average of the links on both sides of a particular chain code member will yield an approximation of the external angle of the shape. A threshold can be used to determine which values correspond to a significant angle change. With a proper selection of the threshold the square and the triangle can be discriminated. The square should have four significant angle changes and the triangle should have three significant angle changes. The circle is identified using a slightly different approach. A digital representation of a circle will

contain on average angle changes of less than $\pi/2$ radians and both the square and triangle will contain at least one exterior angle that is greater than or equivalent to $\pi/2$ radians. Therefore, counting the number of angles greater than or equal to $\pi/2$ will determine if the object is a circle. This is assuming of course that the number of pixels averaged does not exceed 1/8 of the chain code. Of course if the sampling grid is sufficiently coarse it will be impossible to distinquish any of the shapes.

CHAPTER 5
EXPERIMENTAL SECTION

In the sections that follow a labeling convention for the fiducial marks will be utilized. Each mark is labeled with a three letter code, where the first letter of the code is the first letter in the name of the outer shape, the second letter of the code is the first letter in the name of the middle shape, and the third letter of the code is the first letter in the name of the inner shape. For example, the code 'cct' corresponds to the mark that contains a circle as the outer and middle shapes and a triangle as the inner shape.

5.1 Simulation Results

Digital representations of a square, a circle, and an equilateral triangle are created. The shapes are represented initially by a finite set of points. For example, a square would be represented by its four vertices. In the first series of tests, the points are used to obtain a digital representation of the shape, and features are extracted. In particular the moment invariants and the external angles changes are extracted to verify the analytical results. To simulate the results of orthogonal projection, the points are rotated in space and appropriately transformed. The digital representation is obtained, and the features are extracted. This is done to observe the feature changes under these types of transformations. What follows is the data obtained from these simulations.

The first set of simulations consist of rotating the square and equilateral triangle in the image plane from 0 to 180 degrees in increments of 9 degrees. The invariant moments are extracted at every orientation. This is repeated for different scale factors (ratio of image size to pixel size). The scale factors range from 3 to 20. The results for scale factors of 3, 10, and 20 are shown in Tables 5.1 and 5.2. Since

Scale	Maximum	Minimum	mean	standard deviation
3	0.187500	0.152788	0.1836400	0.01091
10	0.166600	0.164931	0.1656439	5.98×10^{-4}
20	0.166875	0.165650	0.1663562	4.34×10^{-4}

Table 5.1: First invariant moment of a rotated and scaled square

Scale	Maximum	Minimum	mean	standard deviation
3	0.2287390	0.173416	0.1915400	0.01765
10	0.197430	0.187741	0.19279755	2.45×10^{-3}
20	0.193099	0.191600	0.19251845	3.60×10^{-4}

Table 5.2: First invariant moment of a rotated and scaled equilateral triangle

there is no change in the digital representation of a circle undergoing rotation in the image plane, it is sufficient to extract the invariant moments once for each scale factor. The results for scale factors of 3, 5, 10, 15, and 20 are shown in Table 5.3

It is clear from Tables 5.1, 5.2, and 5.3 that the moment invariants for a digital image are not strictly invariant, but for the larger scale factors there is a negligible deviation between the theoretical values and the ones obtained via simulation. For example with a scale factor of 20, the percentage error between the theoretical value and the experimental value of ϕ_1 for the square, circle, and equilateral triangle

Scale	ϕ_1
3	.158750
5	.159024
10	.159120
15	.159136
20	.159143

Table 5.3: First invariant moment of a scaled circle

ϕ (degrees)	α (degrees)	ϕ_1	ϕ_2	ϕ_3
27	9	0.167588	0.000502	0.000000
27	27	0.170915	0.001510	0.000000
45	45	0.209200	0.016066	0.000000

Table 5.4: Invariant moments of a rotated and orthogonally projected square

ϕ (degrees)	α (degrees)	ϕ_1	ϕ_2	ϕ_3
27	9	0.200274	0.009220	0.004566
27	27	0.241427	0.02765	0.011028
45	45	0.245370	0.031400	0.005398

Table 5.5: Invariant moments of a rotated and orthogonally projected equilateral triangle

are 0.183, 0.0075, and 0.09 , respectively. It is also clear from the data that the first invariant moments are sufficient to distinquish the shapes obtained via this simulation.

The next series of tests simulate rotation about the x-axis of the object-centered coordinate system by an angle of α and about the y-axis of this system by an angle ϕ, where the values of α and ϕ vary from 9 to 45 degrees in increments of 9 degrees. After rotation, the shapes are orthogonally projected, and the invariant moments are extracted. Tables 5.4 and 5.5 show some of the results. Incidentally, because of the similarity between the invariant moments of the circle and the square, this table was omitted.

It is clear from Tables 5.5 and 5.4 that merely using ϕ_1 as a decision criteria yields incorrect results because there are occasions when the value of ϕ_1 for the orthogonally projected shapes overlap. But the invariant moments can still be used to discriminate between the three shapes. For example from Tables 5.5 and 5.4,

it is readily observed that the third invariant moment is less than 10^{-6} for the square and is greater than 10^{-3} for the equilateral triangle. Similiarily for the circle, the third invariant moment is less than 10^{-6}. This test can be used to uniquely identify the equilateral triangle. To differentiate between the circle and the square, the orthogonal invariant derived in Chapter Four can be utilized. This invariant is given by $\phi_1^2 - \phi_2$. For a continuous square the orthogonal invariant is equivalent to $\frac{1}{6^2}$, and for a continuous circle it is equivalent to $\frac{1}{(2\pi)^2}$. For the discrete representations of the circle and square, the deviation from these values are a function of the scale factor. For a scale factor of thirty, the percentage error obtained from the calculation of the orthogonal invariant for the discrete circle and square is less than 0.3 percent, and for a scale factor of twenty, the percentage error is less than 0.8 percent. As a result, The discrimination of the three digitally represented shapes under orthogonal projection can be obtained.

The simulations that are performed for the moments are repeated using the features based on the ψ-s data, with the notable exception that the scale factor was never reduced below 10. The algorithm performed well. Circles and squares are always correctly classified, and the triangles are misclassified once.

5.2 Experimental Results

A Javelin CCD camera and Data Cube frame grabber are utilized to obtain grey level images of twenty-seven fiducial marks. The marks are placed at various orientations within the field of view. These images are subsequently thresholded to obtain binary images. The borders of each shape within the fiducial marks are extracted. The centroid is obtained from the border of the outer shape and the moment invariants and the chain coded representation of the ψ - s curve are extracted from each of the borders.

	ϕ_1	ϕ_2	ϕ_3
Outer shape	0.261084	0.030719	0.008405
Middle shape	0.214408	0.004788	0.000631
Inner shape	0.326525	0.022299	0.005739

Table 5.6: Invariant moment calculations of fiducial mark ttc

The rationale behind using the moments as features is to obtain an uncomplicated method to recognize the shapes. This is the reason that the approximation of the perspective transformation by an orthogonal projection is utilized. the transformation properties of the moments based on this simplification would make recognition of the shapes relatively simple. Tables 5.6, 5.7, and 5.8 contain the moment invariants of a few typical fiducial mark images. From these tables, it can be clearly observed that the calculated moment invariants do not possess the convenient properties associated with the moments of the orthogonally projected shapes. This observation leads to conclusion that to extract reliable features from the moments is necessary to utilize another methodology. One methodology is to prescribe values for some of the higher order moments (higher than order two) in an attempt to compensate for the effects of the perspective transformation. This does not completely compensate for the effects, but better results are obtained. The major problem with this approach is it becomes necessary to solve nonlinear equations to obtain the values necessary to fix some of the higher order moments. This added complication diminishes the value of using the moments as features. As a result, it is decided to focus on the data obtained from the ψ - s curves to obtain reliable features.

The features utilized to identify the marks are based on the ψ-s curves of the boundary of each shape. These prove to be more reliable than the moments invariants. Two experiments are devised. In the first experiment, The outer shapes

	ϕ_1	ϕ_2	ϕ_3
Outer shape	0.193425	0.000019	0.000071
Middle shape	0.229891	0.011057	0.000982
Inner shape	0.357245	0.018707	0.056625

Table 5.7: Invariant moment calculations of fiducial mark tst

	ϕ_1	ϕ_2	ϕ_3
Outer shape	0.182459	0.000161	0.000064
Middle shape	0.193400	0.009736	0.000233
Inner shape	0.310022	0.025384	0.018699

Table 5.8: Invariant moment calculations of fiducial mark csc

are identified 100 percent of the time, the middle shapes are identified 96.3 percent of the time, and the inner shapes are never identified. The algorithm fails when trying to classify the inner shapes because there is an implicit assumption that the length of a side of a shape is at least twice as large as the sample size utilized in the chain code averaging scheme, and since for boundary lengths of approximately twenty or less the assumption is not correct, the algorithm could not compensate. A second experiment is designed to correctly identify shapes with relatively small boundary lengths. In this experiment, the squares and triangles are discriminated 100 percent of the time, but circles are misclassified. The circles are classified as squares in 7 out of 9 attempts, and the remaining times they are classified as triangles.

Due to poor image quality, there are two occasions when the extraction of a useful middle border is not possible. These cases correspond to the marks 'ttc' and 'stc'. These images are shown in Figures 5.1 and 5.2. Examples of images where the marks are identified are shown in Figures 5.3, 5.4, 5.6 and 5.5. The case where the middle shape is not correctly identified is shown in Figure 5.7. From this figure, the reason the algorithm misclassified the square as a triangle is clear. There is

a significant amount of curvature on one side of the square, and since the external angle change at the intersection of the curved segment and the line segment does not exceed the preset threshold, it is not interpreted as the beginning of a new side. This problem can be eliminated by adjusting the algorithm's threshold. Examples where the algorithm misclassified the inner circles as squares are shown in Figures 5.8 and 5.2. It can be seen from these figures that there exist some chain averaged external angle changes that exceed the preset angle threshold. This causes the algorithm to attempt to classify the shape as either a square or a triangle and it classifies the shape as being closer to a square than a triangle. The algorithm misclassifies the inner circle of fiducial mark 'tcc' in Figure 5.9 as a triangle. This occurs , using the same reasoning as previously mentioned, because the algorithm classifies the shape as being closer to a triangle than a square.

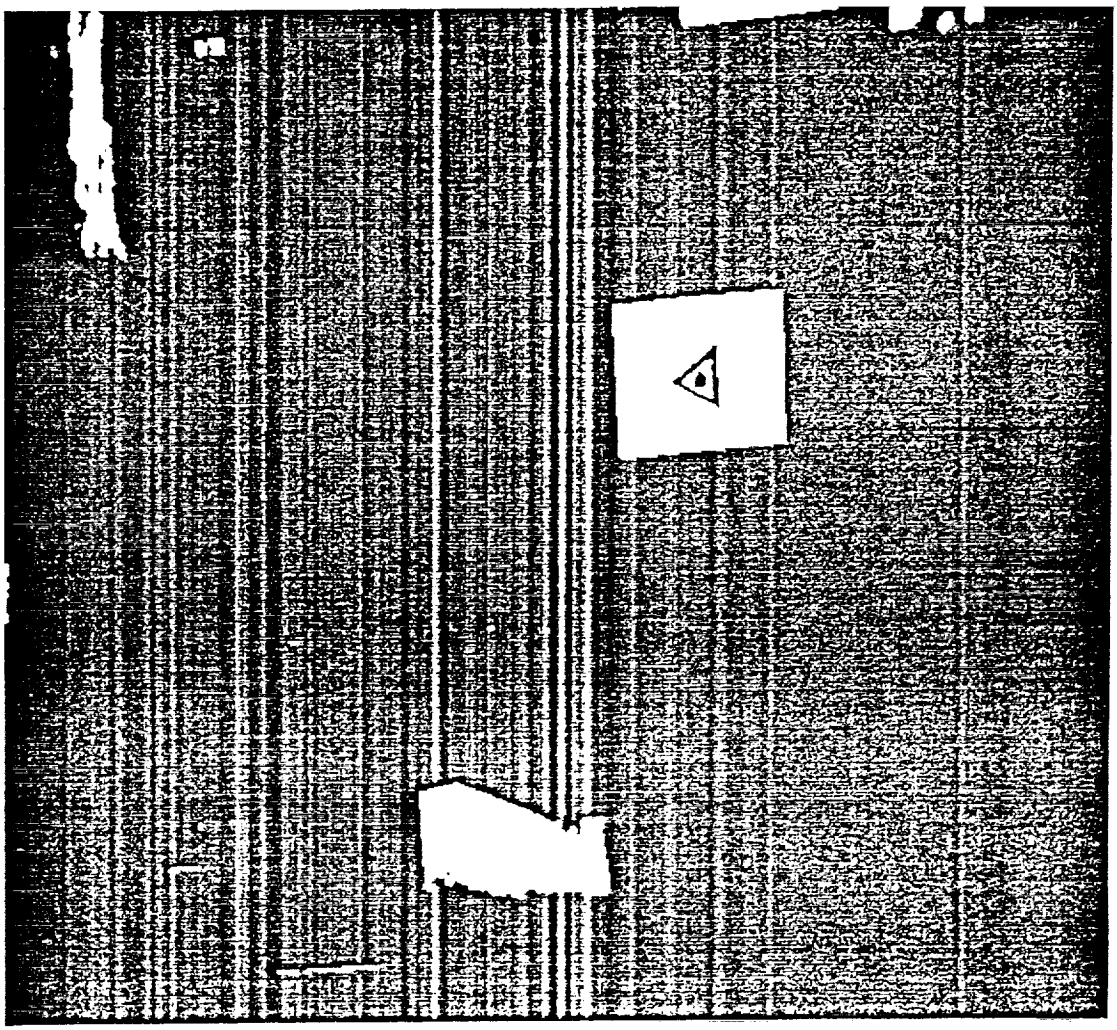

Figure 5.1: Binary image containing fiducial mark ttc

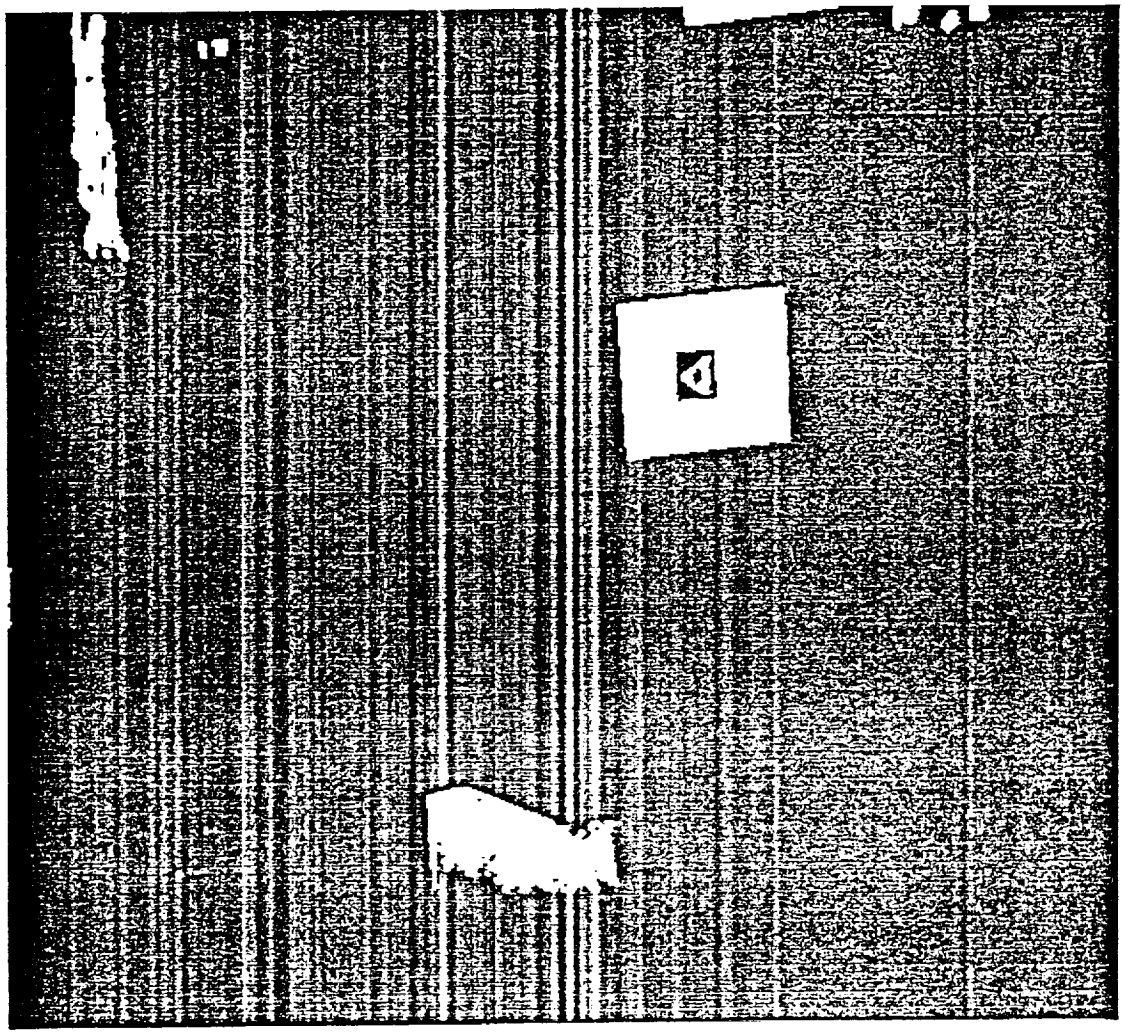

Figure 5.2: Binary image containing fiducial mark stc

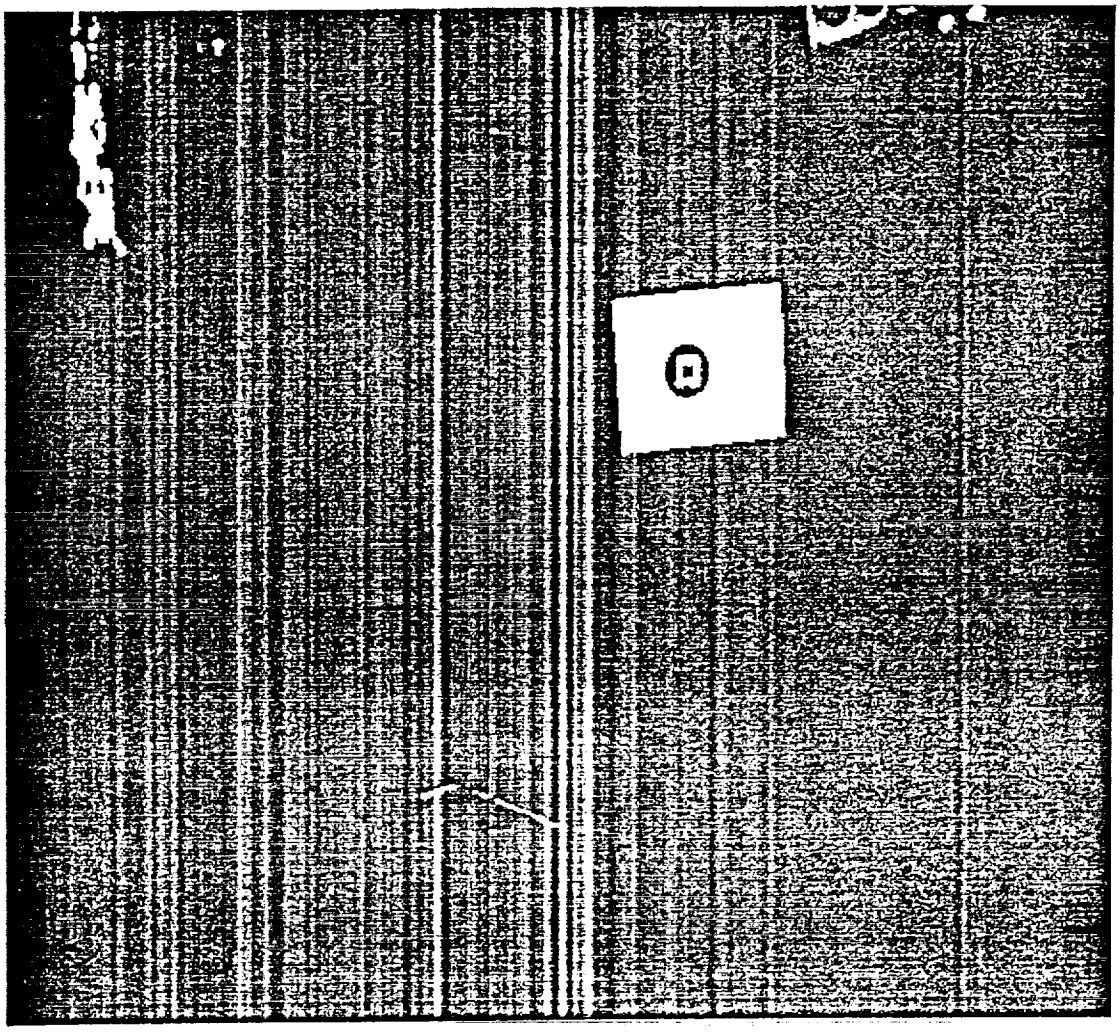

Figure 5.3: Binary image containing fiducial mark css

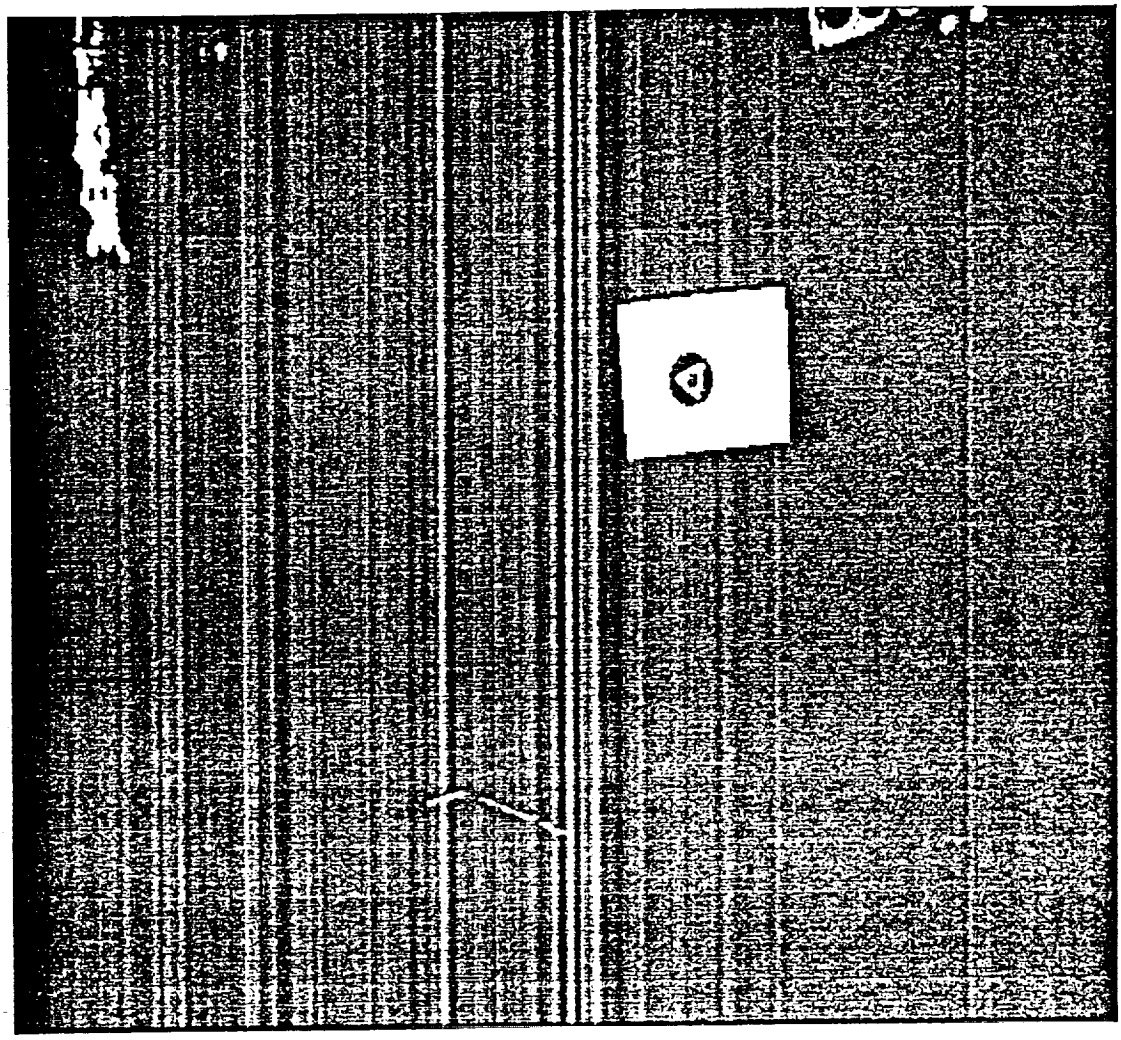

Figure 5.4: Binary image containing fiducial marks

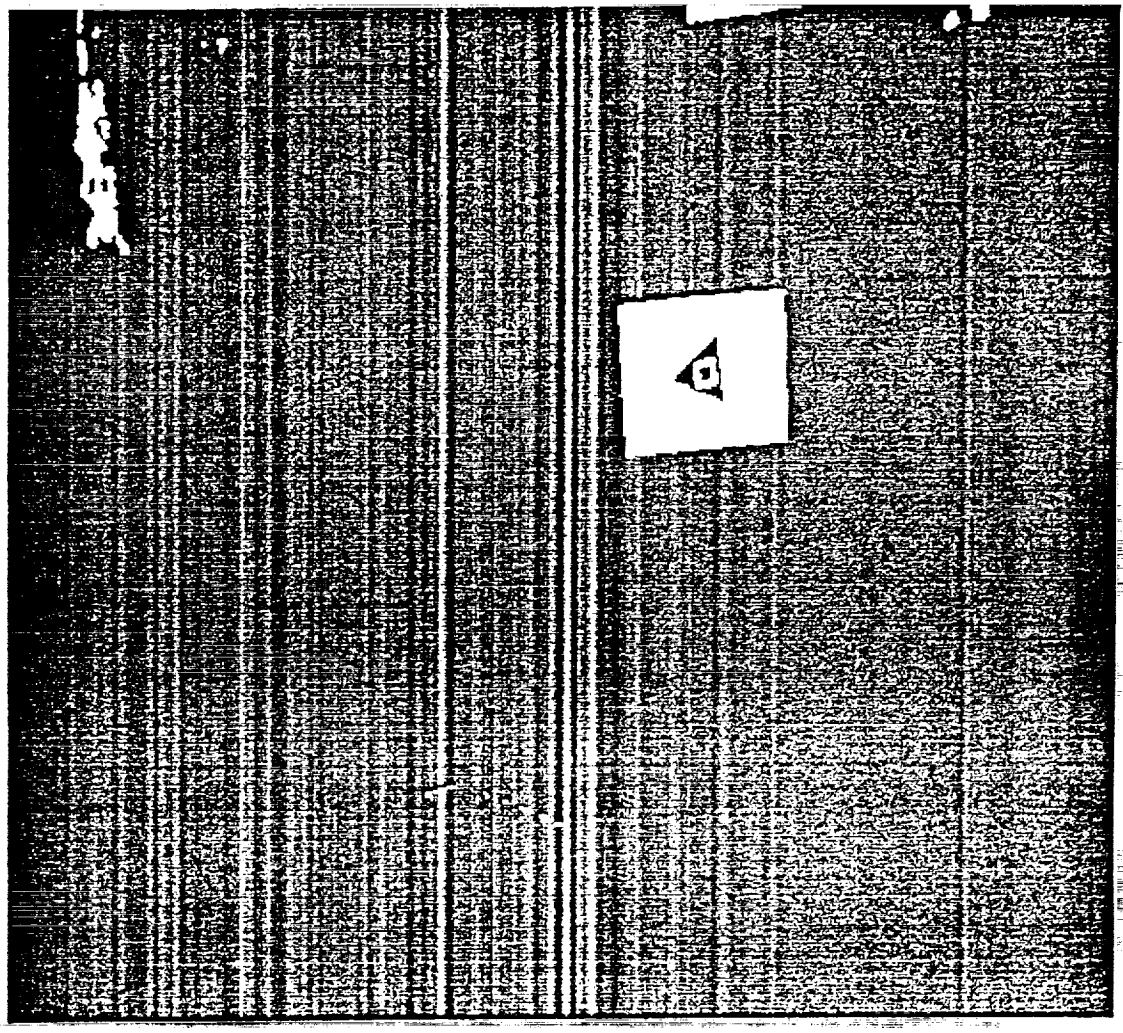

Figure 5.5: Binary image containing fiducial mark tss

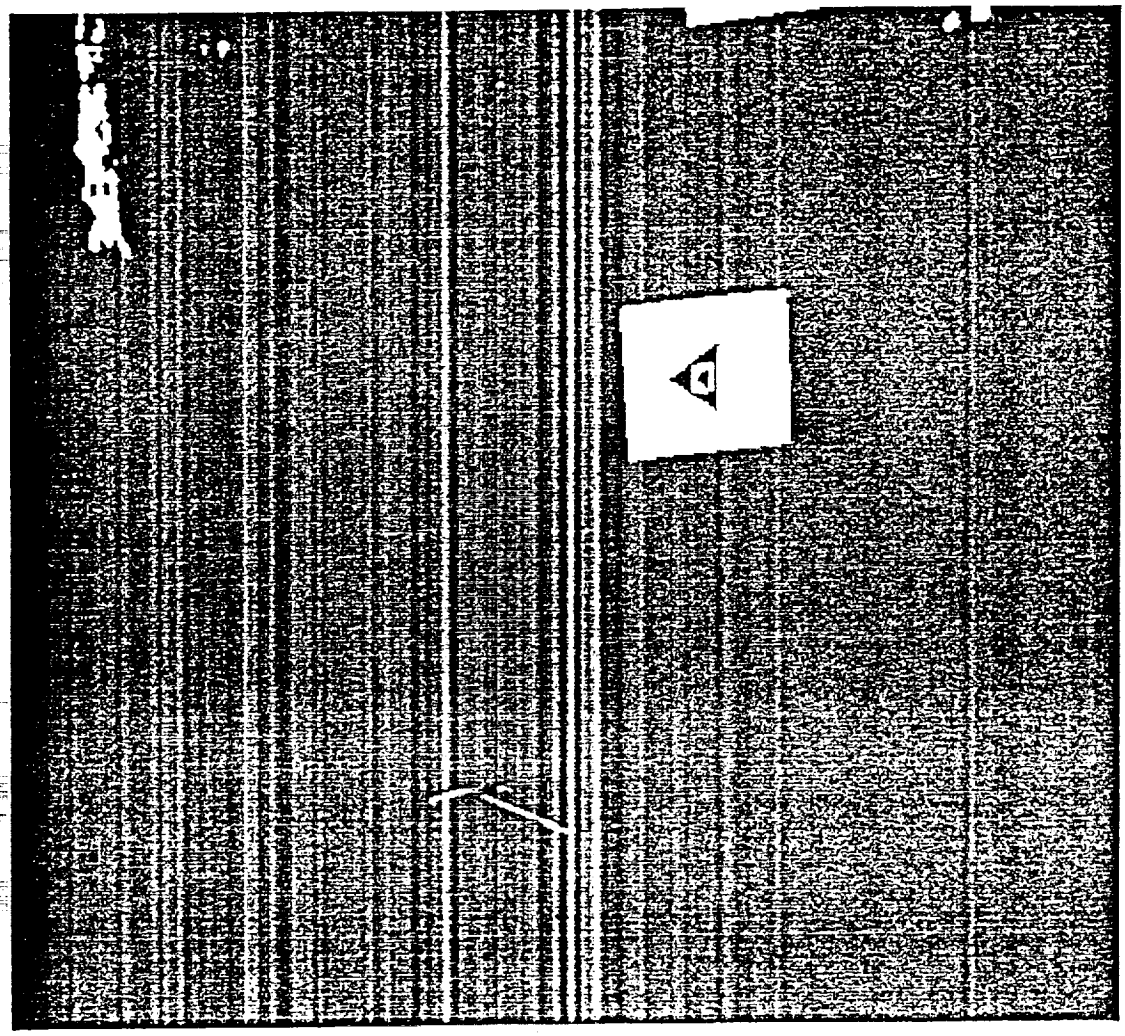

Figure 5.6: Binary image containing fiducial mark tst

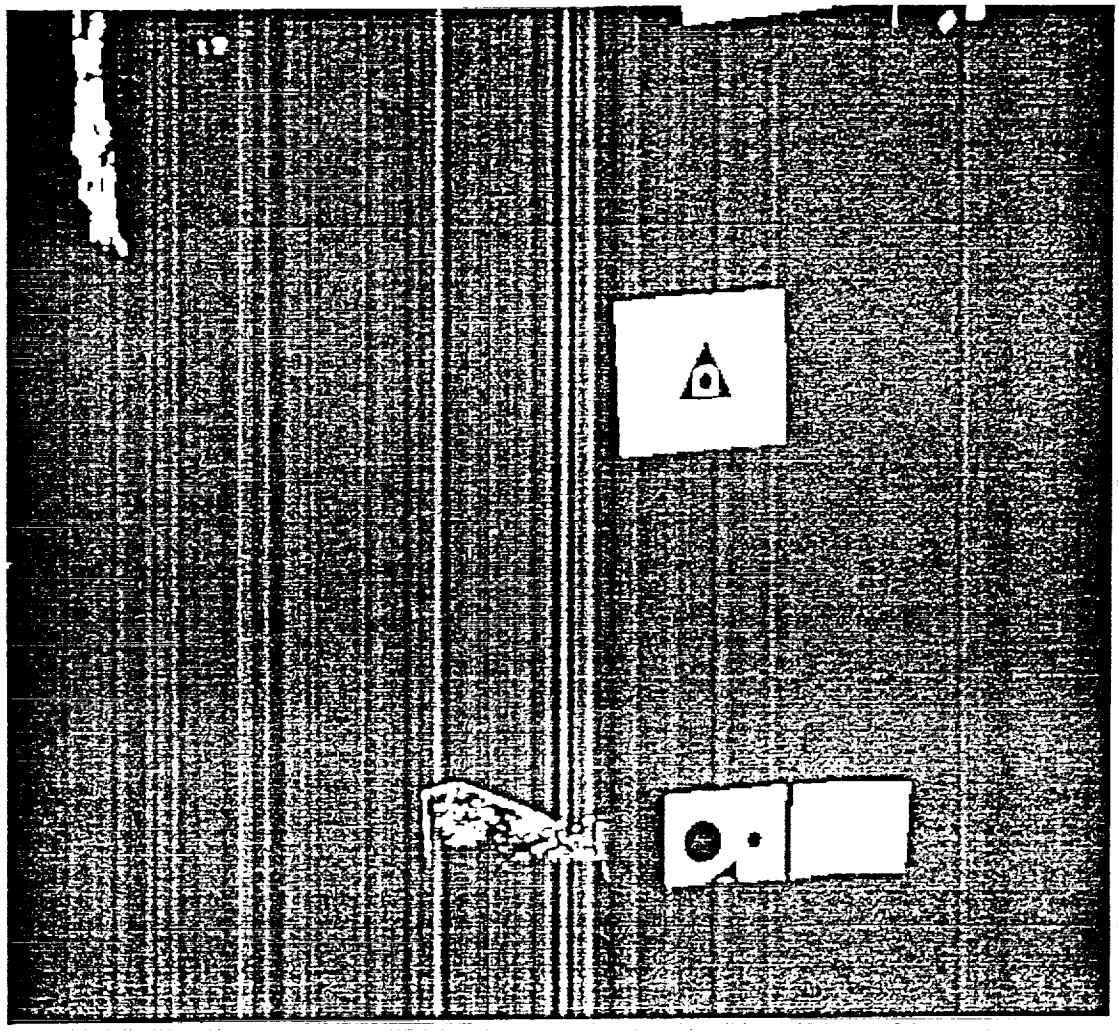

Figure 5.7: Binary image containing fiducial mark tsc

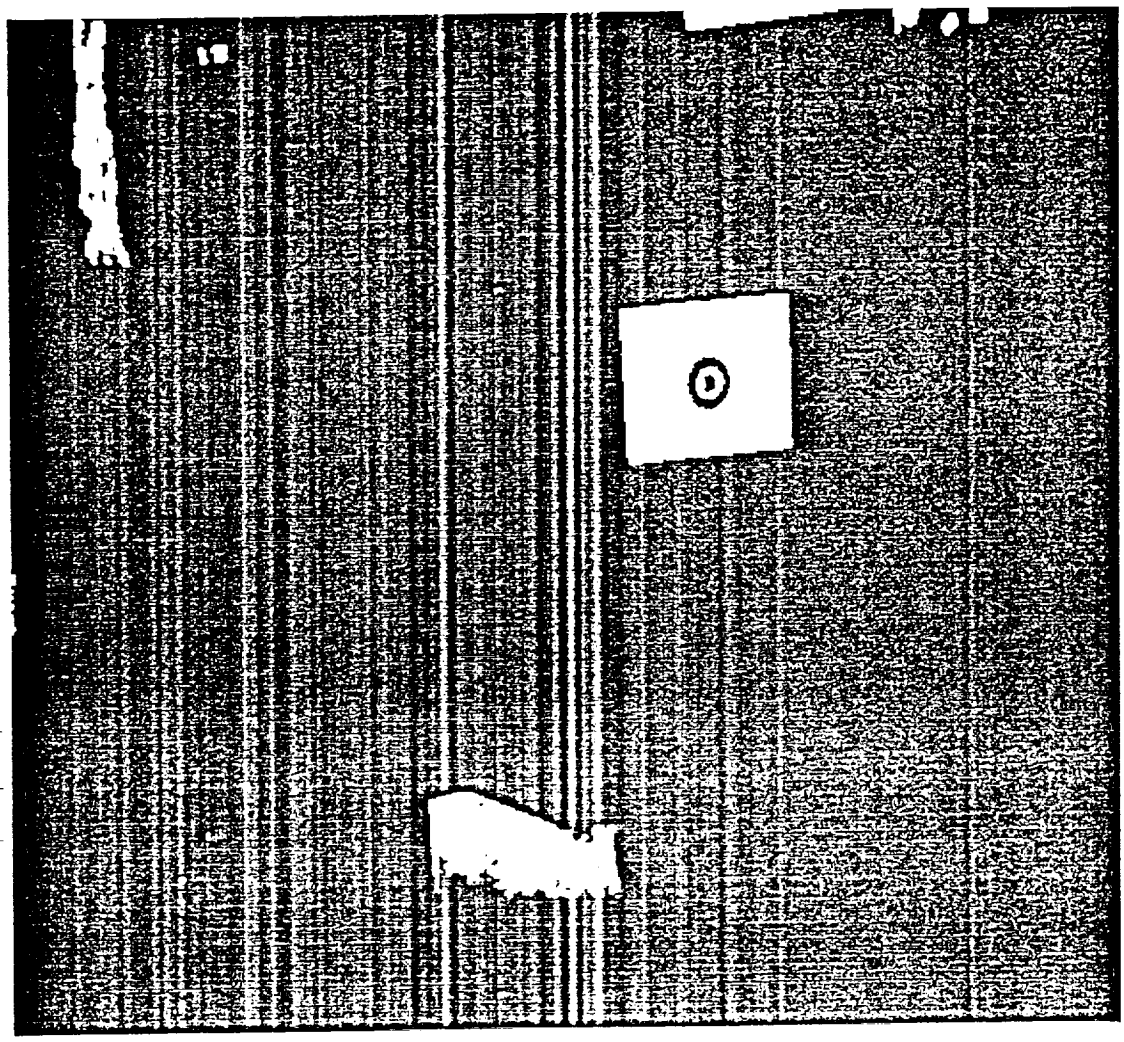

Figure 5.8: Binary image containing fiducial mark ccc

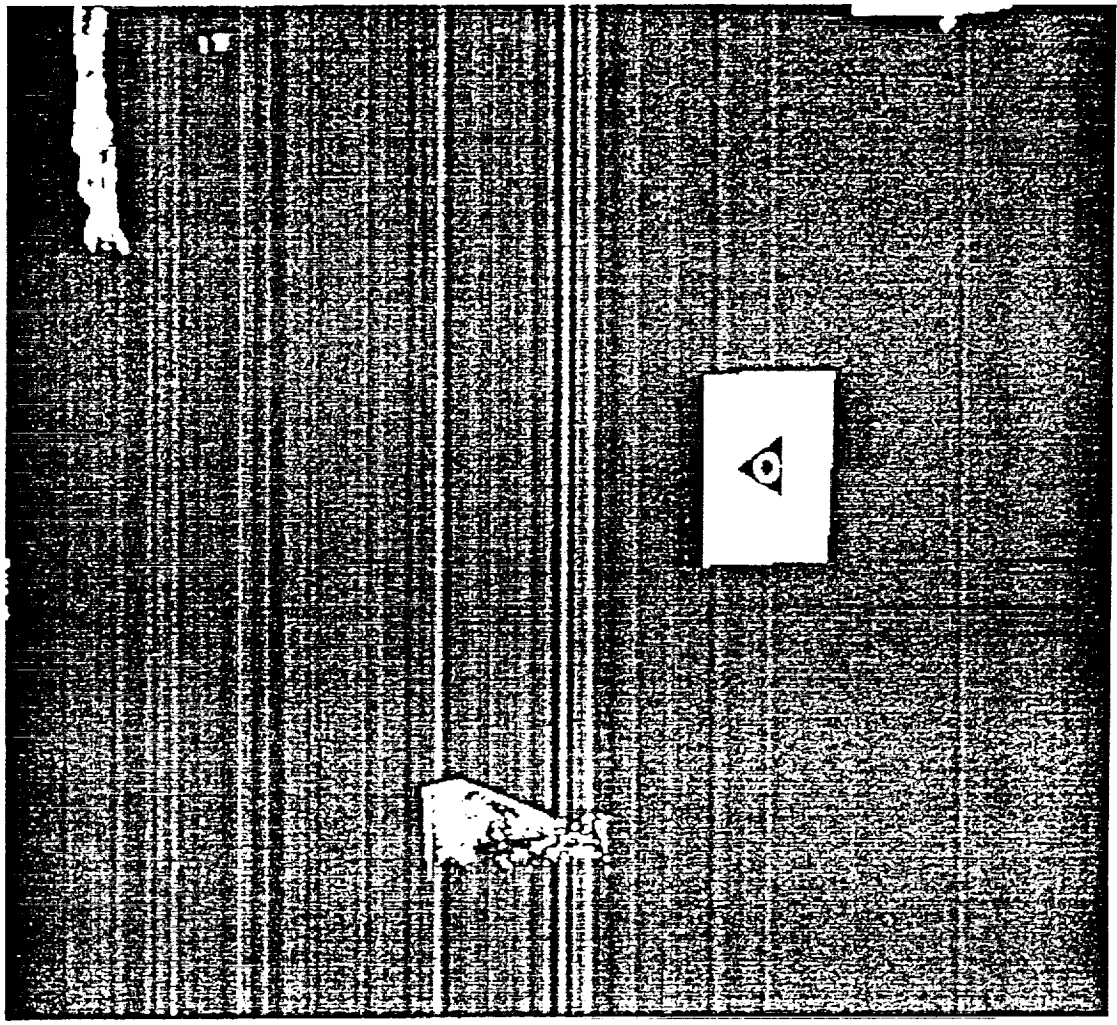

Figure 5.9: Binary image containing fiducial mark tcc

CHAPTER 6
CONCLUSION and FUTURE WORK

A method is proposed to identify the position and orientation of the links of the PUMA 560. To accomplish this task several fiducial marks will be placed on each link. The goal is to uniquely identify and locate each fiducial mark. Two different recognition algorithms are employed in an effort to ascertain the most reliable method to identify the marks.

The first method utilized moment invariants and the second method utilized the chain coded version of the ψ-s curve. For this application, the algorithm based on the ψ-s curve outperforms the algorithm based on the moment invariants. The latter algorithm does not prove to be very robust. The moment algorithm is not able to compensate for the perspective distortion present in the imaged representations of the shapes. The features based on the ψ-s curve performed reasonably well, but the algorithm can be improved.

To find the limitations of the current fiducial mark recognition algorithm, more images will be examined. This can be accomplished by taking images of the fiducial marks at many different angles of inclination with respect to the image plane while constantly varying the distance to the image plane. These images would be used to determine which positions and orientation of the marks cause the algorithm to function poorly. The careful analysis of these results add insight into the search for more robust and efficient algorithms. The tests should encompass the full range of the PUMA 560.

Utilizing the data from the ψ-s curve of the shape boundary, it is possible to obtain more reliable recognition of the marks. Instead of only utilizing the significant angle changes, the entire ψ-s curve can be utilized by segmenting it into straight lines. These straight line segments provide valuable information regarding

the curvature of the boundary. using this methodology, the percentage of the curved portion of the boundary is obtained. This provides an efficient method to discriminate the circle from the other shapes. For example if 70 percent of a boundary is sufficiently curved, the shape is probably a circle. To distinquish between a rectangle and triangle, one can keep track of the positions in the boundary that correspond to the end points of line segments in the ϕ- s curve. This information is used to determine whether a particular line segment in the curve corresponds to a significant portion of the boundary. If it does not then the segment can be interpreted as the result of a poor image.

The improvements in the current algorithm and the verification of the correspondence of the centroid in different views of the same scene are topics that will be explored.

LITERATURE CITED

[1] Hu, M. K. Visual pattern recognition by moment invariants. *IRE Trans. Inform. Theory* **IT-8**: 179-187; 1962.

[2] Udagawa, K.;Toriwaki, J.;Sugino, K. Normalization and recognition of two-dimensional with linear distortion by moments. *Electron Commun. Japan* **47 (6)**: 34-36; 1964.

[3] Alt, F. L. Digital pattern recognition by moments. *J. Ass. Comput. Mach.* **ACM9**: 240-258; 1962.

[4] Casey, R. G. Moment normalization of handprinted characters. *IBM J. Res. Dev.* **14**: 548-557; 1970.

[5] Smith, F. W. Automatic ship photo interpretation by the method of moments. *IEEE Trans. Comput.* **C-20**: 1089-1094; 1971.

[6] Dudani, S. A.; Breeding, K. J.; Mcghee, R. B. Aircraft identification by moment invariants. *IEEE Trans. Comput.* **C-26**: 39-45; 1977.

[7] Teague, M. R. Image analysis via the general theory of moments. *J. Opt. Soc. Am.* **70**: 920-930; 1980.

[8] Wong, R. Y.; Hall, E. L. Scene matching with invariant moments. *Comput. Graphics Image Process.* **8**: 16-24; 1978.

[9] Boyce, J. F.; Hossack, W. J. Moments invariants for pattern recognition. *Pattern Recognition Lett.* **1**: 451-456; 1983.

[10] Khotanzad, A.; Hong Y. H. Rotation invariant image recognition using features selected via a systematic method. *Pattern Recognition.* **23 (10)**: 1089-1100; 1990.

[11] Teh, C.-H.; Chin. R. T. On digital approximation of moment invariants. *Comput. Vision. Graphics. Image Processing* **33**: 318-326; 1986.

[12] Fu, K.S.; Gonzalez R. C.; Lee C.S.G. *Robotics: Control, Sensing, Vision and Intelligence*. McGraw Hill. 1987.

[13] Schalkoff, R.J. *Digital Image Processing and Computer Vision,* Wiley, 1989.