*Published in* **Geometric Invariance in Computer Vision**, *J.L. Mundy and A.Zisserman, eds., MIT Press, pp. 375-397, 1992*

# Object Recognition Based on Moment (or Algebraic) Invariants

Gabriel Taubin* David B. Cooper†

## Abstract

Toward the development of an object recognition and positioning system, able to deal with arbitrary shaped objects in cluttered environments, we introduce methods for checking the match of two arbitrary curves in 2D or surfaces in 3D, when each of these subobjects (i.e., regions) is in arbitrary position, and we also show how to efficiently compute explicit expressions for the coordinate transformation which makes two matching subobjects (i.e., regions) coincide. This is to be used for comparing an arbitrarily positioned subobject of sensed data with objects in a data base, where each stored object is described in some "standard" position. In both cases, matching and positioning, results are *invariant* with respect to viewer coordinate system, i.e., invariant to the arbitrary location and orientation of the object in the data set, or, more generally, to affine transformations of the objects in the data set, which means translation, rotation, and different stretchings in two (or three) directions, and these techniques apply to both 2D and 3D problems. The 3D Euclidean case is useful for the recognition and positioning of solid objects from range data, and the 2D affine case for the recognition and positioning of solid objects from projections, e.g., from curves in a single image, and in motion estimation.

The matching of arbitrarily shaped regions is done by computing for each region a vector of centered moments. These vectors are viewpoint-dependent, but the dependence on the viewpoint is algebraic and well known. We then compute moment invariants, i.e., algebraic functions of the moments that are invariant to Euclidean or affine transformations of the data set. We present a new family of computationally efficient algorithms, based on matrix computations, for the evaluation of both Euclidean and affine algebraic moment invariants of data sets. The use of moment invariants greatly reduces the computation required for the matching, and hence initial object recognition. The approach to determining and computing these moment invariants is different than those used by the vision community previously.

The method for computing the coordinate transformation which makes the two matching regions coincide provides an estimate of object position. The estimation of the matching transformation is based on the same matrix computation techniques introduced for the computation of invariants, it involves simple manipulations of the moment vectors, it neither requires costly iterative methods, nor going back to the data set. The use of geometric invariants in this application is equivalent to specifying a center and an orientation for an arbitrary data constellation in a region.

These geometric invariant methods appear to be very important for dealing with the situation of a large number of different possible objects in the presence of occlusion and clutter. As we point out in this paper, each moment invariant also defines an algebraic invariant, i.e., an invariant algebraic function of the coefficients of the best fitting polynomial to the data. Hence, this paper also introduces a new design and computation approach to algebraic invariants.

## 1 Introduction

In this paper we describe certain aspects of a moment-based approach to 2D and 3D object recognition and positioning in cluttered environments. The data set is either an edge map or 3D range data. This work complements that in an earlier paper [Taubin and Cooper 1990] where the design and computation of algebraic curve and surface invariants was treated. Those invariants are functions of the coefficients of the polynomials used to represent curves in 2D and surfaces in 3D. Central to our approach is the use of geometric invariants. These are used for the fast classification of a region of the data set among a database of regions of known objects, i.e., the matching of a region of the data set with a region in a data base of regions for each of the objects that may possibly be present, invariantly with respect to viewer coordinate system. Then, given a pair of matching regions, one a subset of the data set, and the other a subset of a known object from the database, we show how to efficiently compute the coordinate transformation which makes the two regions coincide. The techniques described in this paper apply to both 2D and 3D problems, under either Euclidean or affine transformations. The 3D Euclidean case is for use in the recognition and positioning of solid objects from range data. Here, the object in the sensed data is a rotation and translation of the object stored in *standard* position in the data base. The 2D affine case is for use in the recognition and positioning of solid objects from projections, and in motion estimation. Here, the curve structure used in the image is assumed to be a view of edges that lie roughly in a plane in three space. For example, this describes the situation of the imaging from the air of an aircraft on the ground. In this situation, the boundaries of the wings, elevators, and fuselage , are seen as though they lie roughly in a plane. Also, surface intersections such as the intersections of the wings and the fuselage are seen in the image as curves that are views of 3D space curves that lie in roughly the same plane as do the aircraft boundary points. Then, when the distance from object to camera is much large than the object diameter, the curves seen in the image are roughly a translation, rotation, and stretching in two directions, of curves lying in a plane in *standard* position in the data base, hence, an affine transformation.

This paper makes a number of contributions of a geometric nature. First, we develop computationally efficient techniques for evaluating moment invariants of finite or continuous sets of points with respect to both Euclidean and affine transformations. We call these sets of points *shapes*. A unified formulation for both the Euclidean and the affine transformations based on vector and matrix techniques is developed for the

*Exploratory Computer Vision Group, IBM T.J.Watson Research Center, P.O.Box 704, Yorktown Heights, NY 10598

†Laboratory For Engineering Man/Machine Systems, Division of Engineering, Brown University, Providence, RI 02912

purpose. Hence, the formulation is more readily accessible to the vision community than are other approaches based on tensor analysis and other techniques. Our results include all previously used invariants as special cases, and new ones as well. The computation used here is based on the finding of eigenvalues, and is significantly less than that required for many of the other previously developed approaches. Second, we develop algorithms for the computation of intrinsic Euclidean and affine coordinate systems of a shape, i.e., a Euclidean or affine normalization procedure. This intrinsic coordinate system is obtained directly from a vector of moments of the shape, and it is independent of the viewer coordinate system, in the sense that that the components of the vector of moments of the same data set in its intrinsic coordinate system are independent of the viewer coordinate system, i.e., they are Euclidean, or affine, invariants. The main application of the affine normalization of shapes is in the recognition and positioning of objects from the projections of some of their contours onto the two-dimensional image plane. This is so because when the camera is far away from the scene, the projective transformation which corresponds to the imaging operation can be approximated by an affine transformation. Although the transformations involved in the projections of 3D objects onto the 2D image plane are projective, we can not consider general projective transformations because they transform bounded sets into unbounded sets, and so, the moments are not well defined with respect to all the projective coordinate systems. Also, the transformation rules of moments with respect to projective transformations are no longer linear. Finally, we emphasize the *computational* aspect of these processes, which are based on both symbolic computations, and well known, efficient, and numerically stable matrix algorithms.

The paper is organized as follows. In section 2 we show how to properly define moments for continuous and discrete data sets. In section 3 we establish the relations between moments computed in different coordinate systems, and show that the design of moment invariants is equivalent to the design of algebraic invariants, i.e., invariants that are functions of the coefficients of the polynomials that represent curves or surfaces in 3D. Hence, our moment invariants trivially determine algebraic invariants.

In section 4 we define and show how to construct covariant and contravariant matrices from moments. Euclidean and affine moment invariants can be obtained by applying matrix operations, such as the evaluation of the determinant or the computation of eigenvalues, to these matrices. In section 5 we show how to compute Euclidean and affine intrinsic coordinate systems from certain covariant and contravariant matrices of a data set, extending the results of section 4. In section 7 we show some simple examples of applications of the methods described in previous sections. In section 6 describe how we intend to use the methods introduced in this paper. In section 8 we review the literature on moment invariants. In section 9 we present our conclusions. And finally, in the appendix (section 10) we give the proofs of the lemmas stated in section 3.

## 2 Data sets, monomials, moments, and invariants

We will consider dense and sparse data sets. Dense data sets are those provided in 3D by laser range scanners, or in 2D by edge detectors. Sparse data sets are composed of easily distinguishable feature points, such as sharp corners, and points of high curvature, which can also be recovered using stereo techniques, or even data provided by tactile sensors.

### 2.1 Moments of finite and continuous data sets

In the case of dense data sets it is better to assume that the data is a sampled version of a $n$-dimensional nonnegative integrable density function $\mu(x)$, and base the analysis on the continuous case. We will only consider density functions which are bounded, nonnegative, and have compact support. In this way, for every polynomial $\phi(x)$, the integral

$$\int \phi \, d\mu \;=\; \int \phi(x)\mu(x)\, dx \tag{1}$$

is finite. Furthermore, we will also require the *total mass* of $\mu$

$$|\mu| \;=\; \int d\mu \;=\; \int \mu(x)\, dx \tag{2}$$

to be positive, otherwise the integral (1) is zero for every polynomial $\phi$. In the case of sparse data sets, a finite set $\{p_1,\ldots,p_q\}$ of $n$-dimensional points, with an associated set of positive weights $\{\mu_1,\ldots,\mu_q\}$, define a singular measure $\mu$. We can apply the same treatment here, where the integral (1) is replaced by the weighted sum $\sum_{i=1}^{q} \phi(p_i)\,\mu_i$, and the total mass (2) by the sum of weights $|\mu| \;=\; \sum_{i=1}^{q} \mu_i$. In both cases we will refer to the measure $\mu$ as a *shape* [Taubin et al. 1989].

We define the moment of a polynomial $\phi(x)$ with respect to a shape $\mu$ as the *normalized* integral

$$M^{\phi} \;=\; \frac{1}{|\mu|}\int \phi\, d\mu \;=\; \frac{1}{|\mu|}\int \phi(x)\mu(x)\, dx \,.$$

The reason for the normalization is the following. If $\mu(x)$ is an integrable density function over the image plane or over 3D range space, $x' = Ax+b$ is an affine coordinate transformation, and $\mu'(x') = \mu(A^{-1}(x'-b))$ is the density function which describes the data in the new coordinate system, according to the change of variables formula, the total mass of $\mu'$ is equal to $|\mu'| = |A||\mu|$, where $|A|$ is the determinant of the matrix $A$, the Jacobian of the affine transformation $x' = Ax+b$, which measures the ratio of areas in 2D and volumes in 3D of corresponding differential regions in the two data sets. With the same reasoning, if $\phi(x)$ is a polynomial, and we write $\phi'(x') = \phi(A^{-1}(x'-b))$, then we have

$$\int \phi' d\mu' \;== |A| \int \phi d\mu \;.$$

The coefficient $|A|$ is no longer present in the relation between the normalized integrals. Hence, when the approach in this paper is applied to determining affine invariants for 2D curves, it is applied by computing moments for regions bounded by the curves of interest rather than the moment invariants for the curves themselves because the invariants depend on the moments computed as integrals over areas rather than as integrals over curves. An analogous statement applies to determining affine invariants for surfaces in 3D space.

If the measure $\mu$ is a finite set of points, the relations between the sums in the two coordinate systems do not include the term $A$.

Finally, if the measure $\mu$ is singular, and it is distributed along a curve, i.e., the moments are computed as curvilinear integrals, meaningful relations between moments computed in two different coordinate systems can be obtain only for Euclidean transformations, because in this case the change of variables formula introduces a term which depends on the skew of the transformation as a function of orientation. This term is constant only for Euclidean transformations. Hence, moment invariants of 2D curves and 3D surfaces can be computed only in the Euclidean case, and local parameterizations are required for this purpose. If only points are available, and they are not uniformly distributed along the curve (with respect to length), we first compute a piecewise linear approximation of the data and compute the moments by integrating along this parameterized curve. In the case of surfaces, we first approximate the data with a triangulated surface, and then compute the moments by integrating along the triangles.

## 2.2 Monomials and moments

A vector of nonnegative integers $\alpha = (\alpha_1, \ldots, \alpha_n)^t$ is a *multiindex*, its *size* is $|\alpha| = \alpha_1 + \cdots + \alpha_n$, and $\alpha! = \alpha_1! \cdots \alpha_n!$ is a multiindex *factorial*. For every multiindex $\alpha$, the polynomial $x_1^{\alpha_1} \cdots x_n^{\alpha_n}$ is a *monomial* of degree $|\alpha|$ which is denoted $x^\alpha$. There are exactly $h_d = \binom{n+d-1}{n-1} = \binom{n+d-1}{d}$ different multiindices of size $d$, and so, that many monomials of degree $d$. Every monomial $x^\alpha$ defines a corresponding *centered moment*

$$M^\alpha \;=\; \frac{1}{|\mu|} \int (x - \overline{\mu})^\alpha \, d\mu(x) \;, \tag{3}$$

where $\overline{\mu} = \frac{1}{|\mu|} \int x \, d\mu(x)$ is the *mean*, or *center*, of the data set described by $\mu$. Centered moments are translation-invariant. That is, if $x' = x + b$ is a translation, which we will see as a coordinate transformation, and $\mu'(x') = \mu(x' - b)$ is the description of the same data set in the new coordinate system, then $M^\alpha \;=\; (M')^\alpha$.

## 2.3 Moment invariants

Moment invariants are functions of the moments of a shape, which are independent of the coordinate system. More precisely, a function $\mathscr{I}(M)$ of the vector of moments $M = (M^\alpha : |\alpha| \leq d)$ of a data set $\mu$ is a *relative invariant* of weight $w$ of $M$ if

$$\mathscr{I}(M') \;=\; |A|^w \, \mathscr{I}(M) \;,$$

for every nonsingular coordinate transformation $x' = Ax$. A relative invariant of weight zero is an *absolute invariant*. A nontrivial absolute invariant can be obtain from two functionally independent relative invariants. If $\mathscr{I}_1(M)$ and $\mathscr{I}_2(M)$ are relative invariants of weight $w_1 \neq 0$ and $w_2 \neq 0$, and $\mathscr{I}_2(M) \neq 0$ (in which case $\mathscr{I}_2(M') \neq 0$ for every coordinate transformation $x' = Ax$), then $\mathscr{I}(M) = [\mathscr{I}_1(M)]^{w_2} / [\mathscr{I}_2(M)]^{w_1}$ defines an absolute invariant of $M$.

Absolute and relative invariants of polynomials, or *algebraic invariants*, can be defined in a similar way as functions of the coefficients, and it is important to note that algebraic and moment invariants are essentially the same. Lemma 3, stated at the end of next section, provides the connection.

The classical theory of algebraic invariants was developed in the nineteenth century by Boole [Boole 1841–1843], Cayley [Cayley 1889–1897], Clebsh [Clebsh 1872], Elliot [Elliot 1913], Gordan [Gordan and Kerschensteiner 1887], Grace and Young [Grace and Young 1903], Hilbert [Hilbert 1890; Hilbert 1893], Sylvester [Sylvester 1904–1912], and others [Dickson 1914; Salmon 1866], to solve the problem of classification of projective algebraic varieties, i.e., sets of common zeros of several homogeneous polynomials. In this century, the main contributions have been by Weyl [Weyl 1939], Mumford [Mumford 1965] and others [Gurevich 1964; Springer 1977]. The projective coordinate transformations define a relation of equivalence in the family of algebraic varieties, with two varieties being equivalent if one of them can be transformed into the other by a projective transformation. In our case, the family of affine (or Euclidean) transformations define a relation of equivalence in the family of 2D or 3D shapes, in a similar fashion.

The classical approach to the classification problem, as for example the classification of planar algebraic curves defined by a single form $\phi(x)$ of degree $d$ in three variables, is to find a set of relative or absolute invariants, $\{\mathscr{I}_1(\phi), \mathscr{I}_2(\phi), \ldots\}$ whose values determine the class that the form belongs to. One naturally tries to find a minimal family, and Hilbert [Hilbert 1890; Hilbert 1893; Ackerman 1978] proved that there exist a *finite* minimal family of *polynomial* invariants, called *fundamental system* of invariants, such that every other polynomial invariant is equal to an algebraic combination of the members of the fundamental system. But Hilbert's proof is not constructive, and the problem, then, is how to compute a fundamental system of polynomial invariants. Algorithms exist, such as the Straightening Algorithm [Rota and Sturmfels 1989], but they are computationally expensive [White 1989].

Due to the the finiteness of the database, constituted in our case by finite moment vectors $M = (M^\alpha : |\alpha| \leq d)$ of regions of models, and the numerical and measurement errors involved, the classification problem that we have to solve is slightly different. We would like to use a fundamental system of polynomial invariants for this purpose, but it is more important to achieve a low computational cost. We only need a sufficiently long vector $\mathscr{I}(M) = (\mathscr{I}_1(M), \ldots, \mathscr{I}_s(M))^t$ of moment invariants with the *separation* property : for every two different members $M_i$ and $M_j$ of the database $\mathscr{I}(M_i) \neq \mathscr{I}(M_j)$, i.e., there must exist an invariant $\mathscr{I}_k$ such that $\mathscr{I}_k(M_i) \neq \mathscr{I}_k(M_j)$, so that members of different classes are mapped to different points in invariant space. These invariants do not have to be necessarily functionally independent, and furthermore, the fundamental systems of polynomial invariants are usually algebraically dependent, so that, it might be worthless to try to find independent invariants. It is not even necessary to find a vector of minimal dimension, as long as it is finite, has the separation property, and is sufficiently inexpensive to evaluate.

# 3 Linear representations and transformation rules

In order to find a sufficient number of moment invariants, and to develop computationally efficient algorithms to evaluate these invariants, we need to study the transformation rules of moments under linear coordinate transformations, i.e., how two sets of moments of the same shape, but computed with respect to two different coordinate systems, relate to each other. We will arrange the moments into vectors and matrices, and then we will use well known numerical methods to compute invariants of these vectors and matrices.

## 3.1 Vectors and matrices of monomials and moments

Multiindices can be linearly ordered in many different ways. We will only use the (inverse) *lexicographical order*, but the same results can be obtained using other orders. If $\alpha$ and $\beta$ are two multiindices of the same size, we say that $\alpha$ precedes $\beta$, and write $\alpha < \beta$, if for the first index $k$ such that $\alpha_k$ differs from $\beta_k$, we have $\alpha_k > \beta_k$. For example, for multiindices of size 2 in three variables, the lexicographical order is

$$(2,0,0) < (1,1,0) < (1,0,1) < (0,2,0) < (0,1,1) < (0,0,2) .$$

And if $\alpha$ and $\beta$ are multiindices of different sizes, we write $\alpha < \beta$ if the size of $\alpha$ is less than the size of $\beta$.

The set of monomials $\{ x^\alpha / \sqrt{\alpha!} : |\alpha| = d \}$ of degree $d$ lexicographically ordered, define a vector of dimension $h_d$, which we will denote $X_{[d]}(x)$. For example,

$$X_{[3]}(x_1,x_2) \;=\; \left( \tfrac{1}{\sqrt{6}}\; x_1^3 \quad \tfrac{1}{\sqrt{2}}\; x_1^2 x_2 \quad \tfrac{1}{\sqrt{2}}\; x_1 x_2^2 \quad \tfrac{1}{\sqrt{6}}\; x_2^3 \right)^t .$$

For every pair of nonnegative integers $(j,k)$, we will denote $X_{[j,k]}(x)$ the $h_j \times h_k$ matrix $X_{[j]}(x) X^t_{[k]}(x)$. That is, $X_{[j,k]}(x)$ is the matrix defined by the set of monomials $\{ x^{\alpha+\beta} / \sqrt{\alpha!\,\beta!} : |\alpha| = j \,,\, |\beta| = k \}$ of degree $d = j+k$, lexicographically ordered according to two subindices. For example,

$$X_{[2,2]}(x_1,x_2,x_3) \;=\;$$

$$\begin{pmatrix}
\frac{1}{2}\; x_1^4 & \frac{1}{\sqrt{2}}\; x_1^3 x_2^1 & \frac{1}{\sqrt{2}}\; x_1^3 x_3^1 & \frac{1}{2}\; x_1^2 x_2^2 & \frac{1}{\sqrt{2}}\; x_1^2 x_2^1 x_3^1 & \frac{1}{2}\; x_1^2 x_3^2 \\
\frac{1}{\sqrt{2}}\; x_1^3 x_2^1 & x_1^2 x_2^2 & x_1^2 x_2^1 x_3^1 & \frac{1}{\sqrt{2}}\; x_1^1 x_2^3 & x_1^1 x_2^2 x_3^1 & \frac{1}{\sqrt{2}}\; x_1^1 x_2^1 x_3^2 \\
\frac{1}{\sqrt{2}}\; x_1^3 x_3^1 & x_1^2 x_2^1 x_3^1 & x_1^2 x_3^2 & \frac{1}{\sqrt{2}}\; x_1^1 x_2^2 x_3^1 & x_1^1 x_2^1 x_3^2 & \frac{1}{\sqrt{2}}\; x_1^1 x_3^3 \\
\frac{1}{2}\; x_1^2 x_2^2 & \frac{1}{\sqrt{2}}\; x_1^1 x_2^3 & \frac{1}{\sqrt{2}}\; x_1^1 x_2^2 x_3^1 & \frac{1}{2}\; x_2^4 & \frac{1}{\sqrt{2}}\; x_2^3 x_3^1 & \frac{1}{2}\; x_2^2 x_3^2 \\
\frac{1}{\sqrt{2}}\; x_1^2 x_2^1 x_3^1 & x_1^1 x_2^2 x_3^1 & x_1^1 x_2^1 x_3^2 & \frac{1}{\sqrt{2}}\; x_2^3 x_3^1 & x_2^2 x_3^2 & \frac{1}{\sqrt{2}}\; x_2^1 x_3^3 \\
\frac{1}{2}\; x_1^2 x_3^2 & \frac{1}{\sqrt{2}}\; x_1^1 x_2^1 x_3^2 & \frac{1}{\sqrt{2}}\; x_1^1 x_3^3 & \frac{1}{2}\; x_2^2 x_3^2 & \frac{1}{\sqrt{2}}\; x_2^1 x_3^3 & \frac{1}{2}\; x_3^4
\end{pmatrix}$$

We will also give special names to the vectors and matrices of centered moments associated with the vectors and matrices of monomials defined above. In the case of vectors, we will denote

$$M_{[d]} \;=\; \frac{1}{|\mu|} \int X_{[d]}(x-\overline{\mu})\, d\mu(x) \,,$$

and in the case of matrices

$$M_{[j,k]} \;=\; \frac{1}{|\mu|} \int X_{[j,k]}(x-\overline{\mu})\, d\mu(x) \,.$$

$M_{[d]}$ is a vector of dimension $h_d$ and $M_{[j,k]}$ is a $h_j \times h_k$ matrix.

## 3.2 Coordinate transformations and linear representations

If $x' = Ax + b$ is a nonsingular affine coordinate transformation, we will denote by $M'_{[d]}$ and $M'_{[j,k]}$ the corresponding vectors and matrices of moments with respect to the new coordinate system. That is

$$M'_{[d]} \;=\; \frac{1}{|\mu'|} \int X_{[d]}(x' - \overline{\mu'})\, d\mu'(x') \,,$$

and

$$M'_{[j,k]} \;=\; \frac{1}{|\mu'|}\int X_{[j,k]}(x'-\overline{\mu'})\,d\mu'(x')\;,$$

where $\mu'(x') = \mu(A^{-1}(x'-b))$ is the description of the weighted data set in the new coordinate system. Since the moments are centered, these new vectors and matrices are independent of the translation part $b$ of the coordinate transformation. From now on, and without loss of generality, we will assume that the translation part is zero.

We will write polynomials expanded in Taylor series

$$f(x) \;=\; \sum_{\alpha} \tfrac{1}{\alpha!}\,F_\alpha\,x^\alpha\;. \tag{4}$$

The coefficients of $f$ are equal to the partial derivatives of order $d$ evaluated at the origin

$$F_\alpha \;=\; \left.\frac{\partial^{\alpha_1+\cdots+\alpha_n} f}{\partial x_1^{\alpha_1}\cdots\partial x_n^{\alpha_n}}\right|_{x=0}$$

and only finitely many coefficients are different from zero. A polynomial is homogeneous, or a *form*, if every one of its terms is of the same degree

$$\phi(x) \;=\; \sum_{|\alpha|=d} \tfrac{1}{\alpha!}\,\Phi_\alpha\,x^\alpha\;. \tag{5}$$

In particular, the monomials are forms. For example, a fourth degree form in three variables is

$$\begin{array}{lllllllll}
\multicolumn{9}{l}{\phi(x_1,x_2,x_3) \;=} \\
\frac{1}{24}\;\Phi_{(4,0,0)}\;x_1^4 & + & \frac{1}{6}\;\Phi_{(3,1,0)}\;x_1^3x_2 & + & \frac{1}{6}\;\Phi_{(3,0,1)}\;x_1^3x_3 & + \\
\frac{1}{4}\;\Phi_{(2,2,0)}\;x_1^2x_2^2 & + & \frac{1}{2}\;\Phi_{(2,1,1)}\;x_1^2x_2x_3 & + & \frac{1}{4}\;\Phi_{(2,0,2)}\;x_1^2x_3^2 & + \\
\frac{1}{6}\;\Phi_{(1,3,0)}\;x_1x_2^3 & + & \frac{1}{2}\;\Phi_{(1,2,1)}\;x_1x_2^2x_3 & + & \frac{1}{2}\;\Phi_{(1,1,2)}\;x_1x_2x_3^2 & + \\
\frac{1}{6}\;\Phi_{(1,0,3)}\;x_1x_3^3 & + & \frac{1}{24}\;\Phi_{(0,4,0)}\;x_2^4 & + & \frac{1}{6}\;\Phi_{(0,3,1)}\;x_2^3x_3 & + \\
\frac{1}{4}\;\Phi_{(0,2,2)}\;x_2^2x_3^2 & + & \frac{1}{6}\;\Phi_{(0,1,3)}\;x_2x_3^3 & + & \frac{1}{24}\;\Phi_{(0,0,4)}\;x_3^4 & .
\end{array}$$

Alternatively, we can write a form as an inner product of a vector of coefficients times a vector of monomials $\phi(x) = \Phi^t_{[d]}X_{[d]}(x)$, where the vector $\Phi^t_{[d]}$ is the set $\{\,\Phi_\alpha/\sqrt{\alpha!} : |\alpha| = d\}$ ordered lexicographically.

If $x' = Ax$ is a nonsingular linear transformation, for every form $\phi(x)$, the polynomial $\phi(Ax)$ is a form of the same degree. In particular, every component of the vector $X_{[d]}(Ax)$ can be written in a *unique way* as a linear combination of the elements of $X_{[d]}(x)$, or in matrix form

$$X_{[d]}(Ax) \;=\; A_{[d]}X_{[d]}(x)\;,$$

where $A_{[d]}$ is a nonsingular $h_d \times h_d$ matrix. We will call the map $A \mapsto A_{[d]}$ the *d-th. degree representation*, and the matrix $A_{[d]}$ the *d-th. degree representation matrix* of $A$. Furthermore,

**Lemma 1** *The map $A \mapsto A_{[d]}$ satisfies the following properties :*

1. *It defines a* faithful linear representation *(a $1-1$ homomorphism of groups) of the group of nonsingular $n\times n$ matrices $GL(n)$ into the group of nonsingular $h_d\times h_d$ matrices $GL(h_d)$. That is, for every pair of nonsingular matrices $A, B$, we have (a): $(AB)_{[d]} = A_{[d]}B_{[d]}$ (preserves products), (b): if $A_{[d]} = B_{[d]}$, then $A = B$ (is one to one), the matrix $A_{[d]}$ is nonsingular, and (c): $(A_{[d]})^{-1} = (A^{-1})_{[d]}$.*

2. *It preserves transposition, i.e., for every nonsingular matrix $A$, we have $(A^t)_{[d]} = (A_{[d]})^t$. In particular, if $A$ is symmetric, positive definite, or orthogonal, so is $A_{[d]}$.*

3. *If $A$ is lower triangular, so is $A_{[d]}$. In particular, if $A$ is diagonal, so is $A_{[d]}$.*

4. *The determinant of $A_{[d]}$ is equal to $|A|^m$, with $m = \binom{n+d-1}{n-1}$.*

Now, we can establish the transformation rules of moments under coordinate transformations.

**Lemma 2** *For every nonsingular affine transformation $x' = Ax$, and nonnegative integers $d, j, k$, we have*

1. $M'_{[d]} = A_{[d]}M_{[d]}$.

2. $M'_{[j,k]} = A_{[j]}M_{[j,k]}A^t_{[k]}$,

*where the moments are computed as surface (volume) integrals. In the Euclidean case, the same relations hold, but the moments can be computed not only as surface (volume) integrals, but also as curve (surface) integrals.*

And finally, we establish the relation between moment invariants and algebraic invariants.

**Lemma 3** *Let $x' = Ax$ be a nonsingular affine transformation, let $\phi(x) = \Phi^t_{[d]} X_{[d]}(x)$ be a form of degree $d$, and let $\phi'(x') = \phi(A^{-1}x')$. If we write $\phi'(x')$ in vector form $\phi'(x') = \Phi'^t_{[d]} X_{[d]}(x')$, then, $\Phi'_{[d]} = A^{-t}_{[d]} \Phi_{[d]}$.*

Thus, $\mathscr{I}(M_{[d]})$ is an invariant of weight $w$ of the tor of moments $M_{[d]}$, if and only if $\mathscr{I}(\Phi_{[d]})$ is an invariant of weight $-w$ of the vector of coefficients $\Phi_{[d]}$.

# 4 Computing moment invariants

We want to emphasize the computational aspect of the methods described below. For example, the complexity of numerically computing the determinant of a square $n \times n$ matrix $A$ is in the order $n^4$ arithmetic operations, because, in the order of $n^3$ operations are needed for computing the QR decomposition of $A$, and exactly $n-1$ multiplications to compute the determinant of the triangular matrix of the decomposition [Golub and Van Loan 1983]. However, the analytic expression of the determinant $|A|$ as a polynomial of degree $n$ in the $n^2$ elements of the matrix, has $n!$ terms. Some of these techniques have been well known for a century, but our emphasis on structuring the algorithms for the efficient numerical computation of invariants based on matrix computations is new.

## 4.1 Covariant and contravariant matrices

The fundamental relative invariant of the moments of degree $2d$ is the determinant of the matrix $M_{[d,d]}$. If $x' = Ax$ is a coordinate transformation, according to lemmas 1 and 2, we have

$$\left|M'_{[d,d]}\right| = \left|A_{[d]} M_{[d,d]} A^t_{[d]}\right| = \left|A_{[d]}\right|^2 \left|M_{[d,d]}\right| = |A|^{2m} \left|M_{[d,d]}\right| \; ,$$

where $m = \binom{n+d-1}{n-1}$. That is, $|M_{[d,d]}|$ is a relative invariant of weight $2m$. Note that the invariance of $|M_{[d,d]}|$ follows only from the transformation rules of $M_{[d,d]}$, and not from the fact that the elements of the matrix $M_{[d,d]}$ are moments of degree $d$.

In general a matrix $\mathscr{C}_{[j,k]}(M)$ whose components are functions of the moments $M = (M^\alpha : |\alpha| \geq 0)$, and such that

$$\mathscr{C}_{[j,k]}(M') \; = \; A_{[j]} \, \mathscr{C}_{[j,k]}(M) \, A^t_{[k]}$$

will be called *covariant matrix*, and will be briefly denoted $\mathscr{C}_{[j,k]}$, while the same matrix function evaluated in a different coordinate system $\mathscr{C}_{[j,k]}(M')$ will be denoted $\mathscr{C}'_{[j,k]}$. Note that $|\mathscr{C}_{[d,d]}|$ defines a new relative invariant of weight $2m$.

If the matrix $\mathscr{C}_{[j,k]}$ satisfies

$$\mathscr{C}'_{[j,k]} \; = \; A^{-t}_{[j]} \, \mathscr{C}_{[j,k]} \, A^{-1}_{[k]}$$

instead, it will be called *contravariant matrix*. If it satisfies

$$\mathscr{C}'_{[j,k]} \; = \; A_{[j]} \, \mathscr{C}_{[j,k]} \, A^{-1}_{[k]}$$

will be called *left covariant and right contravariant*, with a similar definition for matrices which are *left contravariant and right covariant*. Clearly, the determinant of a square contravariant matrix $\mathscr{C}_{[d,d]}$ is a relative invariant of weight $-2m$, and the determinant of a square left covariant and right contravariant matrix is an absolute invariant.

The simplest example of a square contravariant matrix, which is not a matrix of moments, is $M^{-1}_{[d,d]}$, which is generally well defined, unless a form of degree $d$ interpolates all the data set, making the matrix $M_{[d,d]}$ singular. Otherwise, the matrix $M_{[d,d]}$ is positive definite, and so, invertible.

If the coordinate transformations are restricted to Euclidean transformations, i.e., the matrix $A$ is orthogonal, the four kinds of matrices defined above coincide, and we only talk of covariant matrices. Furthermore, since $A_{[d]}$ is orthogonal when $A$ is orthogonal, a matrix $\mathscr{C}_{[j,k]}$ is covariant with respect to orthogonal transformations if it satisfies

$$\mathscr{C}'_{[j,k]} \; = \; A_{[j]} \, \mathscr{C}_{[j,k]} \, A^t_{[k]} \; .$$

If $\mathscr{C}_{[j,k]}$ is also square, with $j = k = d$, then, its $h_d$ eigenvalues are orthogonal invariants, because in this case, the matrix $\mathscr{C}_{[d,d]} - \theta I$ is also covariant for every value of $\theta$, and so, the coefficients, or equivalently the roots, of the characteristic polynomial $|\mathscr{C}_{[d,d]} - \theta I|$, are invariants. More generally, the eigenvalues of a left covariant and right contravariant matrix are absolute invariants, by a similar argument. Note that, from the computational point of view, computing eigenvalues is much less expensive than expanding the determinants needed to obtain the coefficients of the characteristic polynomials, and computing eigenvalues requires in the order of $n^3$ operations, where $n$ is the size of the square matrices involved.

The results of this section are not the only methods to compute invariants [Taubin 1991b], but due to limited space, we will omit the description of other techniques. However, these are the fundamental tools for the computation of affine and Euclidean moment invariants described below.

### 4.2 Euclidean moment invariants

New Euclidean covariant matrices can be constructed multiplying other Euclidean covariant matrices of proper sizes, or by restricting affine covariant matrices to Euclidean transformations.

If $\mathscr{C}_{[j,k]}$ and $\mathscr{D}_{[k,l]}$ are two Euclidean covariant matrices, then $\mathscr{E}_{[j,l]} = \mathscr{C}_{[j,k]}\mathscr{D}_{[k,l]}$ is also a Euclidean covariant matrix, because

$$\mathscr{E}'_{[j,l]} \;=\; \left[A_{[j]}\mathscr{C}_{[j,k]}A^t_{[k]}\right]\left[A_{[k]}\mathscr{D}_{[k,l]}A^t_{[l]}\right] \;=\; A_{[j]}\left[\mathscr{C}_{[j,k]}\mathscr{D}_{[k,l]}\right]A^t_{[l]}$$

The simplest Euclidean moment invariants are the eigenvalues of the matrix $M_{[1,1]}$, functions of the centered moments of degree $2$. Then we have the eigenvalues of $M_{[2,2]}$, which are functions of the centered moments of degree $4$ and $M_{[1,2]}M_{[2,2]}M_{[2,1]}$ and $M_{[1,2]}M^{-1}_{[2,2]}M_{[2,1]}$, which are functions of the centered moments of degree $3$ and $4$. Join invariants of all the centered moments of degree $2$, $3$, and $4$ can be obtained as the eigenvalues of the block matrix

$$\begin{pmatrix} M_{[1,1]} & M_{[1,2]} \\ M_{[2,1]} & M_{[2,2]} \end{pmatrix} . \tag{6}$$

In general, eigenvalues of block matrices built in this way are always joint Euclidean invariants of the elements of the component matrices. The transformation rules can be easily derived from those of the component matrices. For example

$$\begin{pmatrix} M'_{[1,1]} & M'_{[1,2]} \\ M'_{[2,1]} & M'_{[2,2]} \end{pmatrix} = \begin{pmatrix} A_{[1]} & 0 \\ 0 & A_{[2]} \end{pmatrix}\begin{pmatrix} M_{[1,1]} & M_{[1,2]} \\ M_{[2,1]} & M_{[2,2]} \end{pmatrix}\begin{pmatrix} A_{[1]} & 0 \\ 0 & A_{[2]} \end{pmatrix}^t .$$

Another family of Euclidean covariant matrices is defined by the following formula

$$N_{[j,k]} \;=\; \frac{1}{|\mu|}\int [DX_{[j]}(x-\overline{\mu})][DX_{[k]}(x-\overline{\mu})]^t\, d\mu(x) \,,$$

where $DX_{[j]}(x)$ is the Jacobian matrix corresponding to the vector of monomials $X_{[j]}(x)$. For example,

$$DX_{[2]}(x_1,x_2,x_3) \;=\; \begin{pmatrix} \sqrt{2}x_1 & 0 & 0 \\ x_2 & x_1 & 0 \\ x_3 & 0 & x_1 \\ 0 & \sqrt{2}x_1 & 0 \\ 0 & x_3 & x_2 \\ 0 & 0 & \sqrt{2}x_3 \end{pmatrix} .$$

The matrices $N_{[j,k]}$ are also covariant with respect to Euclidean transformations, but we will omit the proof. Note that the elements of the Jacobian matrix $DX_{[j]}(x)$ are monomials of degree $j-1$, and since $DX_{[j]}(x)$ has $n$ columns, the elements of the matrix $N_{[j,k]}$ are sums of at most $n$ moments of degree $j+k-2$. The matrices $M_{[j,k]}$ and $N_{[j,k]}$ are closely related to the problem of fitting algebraic curves and surfaces to data. If we denote $X(x)$ the vector of monomials of degree $\leq k$, i.e., $X^t = (X^t_{[0]}, X^t_{[1]}, \ldots, X^t_{[k]})$,

$$M \;=\; \frac{1}{|\mu|}\int X(x-\overline{\mu})X(x-\overline{\mu})^t d\mu(x) \;=\; \begin{pmatrix} M_{[0,0]} & \cdots & M_{[0,k]} \\ \vdots & \ddots & \vdots \\ M_{[k,0]} & \cdots & M_{[k,k]} \end{pmatrix}$$

and

$$N \;=\; \frac{1}{|\mu|}\int DX(x-\overline{\mu})DX(x-\overline{\mu})^t d\mu(x) \;=\; \begin{pmatrix} N_{[0,0]} & \cdots & N_{[0,k]} \\ \vdots & \ddots & \vdots \\ N_{[k,0]} & \cdots & N_{[k,k]} \end{pmatrix} ,$$

then, the eigenvector of the pencil $M-\lambda N$ associated with the minimum generalized eigenvalue, corresponds to the vector $F$ of coefficients of the polynomial $f(x) = F^t X(x)$ of degree $\leq k$ such that its set of zeros $\{x : f(x) = 0\}$ best fits the data in the least squares sense [Taubin 1988a; Taubin 1988b; Taubin 1991a].

After transforming every polynomial into a form by switching to homogeneous coordinates, the eigenvector $\Phi_{[k]}$ corresponding to the minimum generalized eigenvalue of the pencil $M_{[k,k]} - \lambda N_{[k,k]}$ is the vector of coefficients of the form $\phi(x) = \Phi^t_{[k]}X_{[k]}(x)$ of degree $k$ whose associated set of zeros $\{x : \phi(x) = 0\}$ approximately best fits the data in the least squares sense.

Other method for computing Euclidean moment invariants, not described in this paper, is the harmonic decomposition [Taubin 1991b], which yields a complete system of invariants, using similar matrix computation techniques.

### 4.3 Affine moment invariants

Eigenvalues of matrices which are covariant on one side and contravariant on the other side are absolute invariants. In this section we show methods to construct such matrices from the moments of a shape, reducing the problem of computing affine moment invariants to the computation of eigenvalues of square matrices, as in the Euclidean case.

The matrices $M_{[j,k]}$ are covariant on both sides, but since the square matrices $M_{[k,k]}$ are usually positive definite, we can define a new family of matrices with the desired properties. For every pair of nonnegative integers $j$ and $k$ we will write

$$H_{[j,k]} \;=\; M_{[j,k]}M_{[k,k]}^{-1} \,. \tag{7}$$

Note that this matrices only make sense for $j \neq k$, because $H_{[k,k]}$ is the identity matrix. Also note that $H_{[j,k]} \neq H_{[k,j]}^t$. Now, if $M_{[j,j]}$ and $M_{[k,k]}$ are positive definite, the square $h_k \times h_k$ matrix

$$H_{[k,j]}H_{[j,k]} \;=\; M_{[k,j]}M_{[j,j]}^{-1}M_{[j,k]}M_{[k,k]}^{-1}$$

is left covariant and right contravariant, and so, its $h_k$ eigenvalues are joint absolute invariants of the moments of degrees $2j, j+k$ and $2k$ under affine transformations. This only makes sense for $k \leq j$, because the other combination yields the same principal values, followed by zeros. For example, if $k=1$ and $j=2$ we obtain the simplest absolute affine moment invariants of a shape, the $h_1 = n$ principal values of the $n\times n$ matrix

$$H_{[1,2]}H_{[2,1]} \;=\; M_{[1,2]}M_{[2,2]}^{-1}M_{[2,1]}M_{[1,1]}^{-1} \,,$$

which is a rational function of the centered moments of degree $2,3$ and $4$.

Another important family of left covariant and right contravariant matrices can be constructed replacing $M_{[k,k]}$ by $M_{[1,1][k]}$ in (7), where $M_{[1,1][k]}$ $k$-th. degree representation of $M_{[1,1]}$, considered as the matrix associated to a coordinate transformation $x' = M_{[1,1]}x$. $M_{[1,1][k]}$ is symmetric and positive definite when $M_{[1,1]}$ is positive definite. In this case we write

$$U_{[j,k]} \;=\; M_{[j,k]}M_{[1,1][k]}^{-1} \,.$$

When $j=k$, the square matrix $U_{[k,k]}$ is no longer the identity matrix, and its $h_k$ eigenvalues are absolute invariants of the shape $\mu$. Since $M_{[1,1]}$ is positive definite, it has a nonsingular square root, a square matrix $L$ such that $LM_{[1,1]}L^t = I$. We can take $L$ as the inverse of the lower triangular Cholesky decomposition of $M_{[1,1]}$. From the properties of the representation map described in Lemma 1, we have

$$M_{[1,1][k]} \;=\; L_{[k]}^{-1}L_{[k]}^{-t}.$$

The matrices $U_{[k,k]}$ and $L_{[k]}U_{[k,k]}L_{[k]}^{-1}$ are conjugate, and so, they have the same characteristic polynomials. However, the last matrix is symmetric

$$L_{[k]}U_{[k,k]}L_{[k]}^{-1} \;=\; L_{[k]}M_{[k,k]}L_{[k]}^t \,,$$

and has all real eigenvalues, which are absolute invariants of the shape $\mu$. The simplest of these matrices corresponds to the case $k=2$, which produces $h_2 = \binom{n+2-1}{n-1} = n(n+1)/2$ absolute invariants, functions of the centered moments of degree $2$ and $4$. The simplest affine absolute invariants of a shape are the $n$ eigenvalues of the $n\times n$ symmetric matrix

$$LU_{[1,2]}U_{[2,1]}L^{-1} \;=\; LM_{[1,2]}M_{[1,1][2]}^{-1}M_{[2,1]}L^t \,,$$

which are functions of the centered moments of degree $2$ and $3$. Note that, if we consider the coordinate transformation $x' = Lx$, this last matrix is nothing but

$$LM_{[1,2]}M_{[1,1][2]}^{-1}M_{[2,1]}L^t = \left(LM_{[1,2]}L_{[k]}^t\right)\left(L_{[k]}M_{[2,1]}L^t\right) = M'_{[1,2]}M'_{[2,1]} \,.$$

This property is the basis for the definition of the intrinsic affine frame of reference of a shape, described in detail in the next section. The coordinate transformation $x' = Lx$ is a normalization which reduces the affine invariants in the original coordinates ($x$) to Euclidean invariants in the new coordinates ($x'$).

## 5 Canonical frame of reference

In this section we are concerned with methods to normalize a shape with respect to Euclidean and affine transformations. We have defined the center of a shape as the mean of the data, the vector of first degree moments, which is an affine covariant vector of the shape. In order to define an intrinsic frame of reference, we still have to determine a canonical orthogonal matrix, in the Euclidean case, and a canonical nonsingular matrix, in the affine case.

As an illustration, the eigenvectors of the matrix of $M_{[1,1]}$ of second order central moments of a data set in 2D or 3D are covariant with respect to the choice of orthogonal coordinate system for the data. Hence these eigenvectors and the center vector determine the unit coordinate vectors and the origin of an intrinsic coordinate system for the data. In this section we generalize this idea.

### 5.1 Euclidean case

In this case we can define the orientation of a shape as one of the $2^n$ orthonormal sets which diagonalizes a symmetric $n\times n$ covariant matrix with nonrepeated eigenvalues. The simplest $n\times n$ covariant matrix is the *scatter matrix* $M_{[1,1]}$, but if this matrix has repeated eigenvalues, we can also use any one of the following matrices

$$\begin{cases} M_{[1,k]}M_{[k,1]} \\ M_{[1,k]}M_{[k,k]}M_{[k,1]} \\ M_{[1,k]}M^{-1}_{[k,k]}M_{[k,1]} \\ M_{[1,k]}M_{[1,1][k]}M_{[k,1]} \\ M_{[1,k]}M^{-1}_{[1,1][k]}M_{[k,1]} \end{cases} \qquad k=2,3\ldots \tag{8}$$

or a linear combination of them

$$\theta_1 M_{[1,1]} + \sum_{k\geq 2} \theta_k \left(M_{[1,k]}M_{[k,1]}\right) + \cdots \ .$$

In order to disambiguate among the $2^n$ candidate orthogonal frames of reference, we use covariant $n$-dimensional vectors. A *covariant vector* is a vector function of moments $v(M)$, such that $v(M') = Av(M)$ for every coordinate transformation $x' = Ax$. Every nonzero element of a covariant vectors can be used to choose the orientation of the corresponding coordinate axis. The simplest covariant vector is $M_{[1]}$, but since the moments are centered, $M_{[1]}$ is identically zero. Other Euclidean covariant vectors can be computed as in (8), as follows

$$\begin{cases} M_{[1,k]}M_{[k]} \\ M_{[1,k]}M_{[k,k]}M_{[k]} \\ M_{[1,k]}M^{-1}_{[k,k]}M_{[k]} \\ M_{[1,k]}M_{[1,1][k]}M_{[k]} \\ M_{[1,k]}M^{-1}_{[1,1][k]}M_{[k]} \end{cases} \qquad k=2,3,\ldots \tag{9}$$

The simplest Euclidean covariant vector of this family is a function of the centered moments of degree $2$ and $3$

$$v_1 \ = \ M_{[1,2]}M_{[2]} \ .$$

If this vector is not identically zero, the following vector is another covariant vector

$$v_2 \ = \ M_{[1,1]}v_1 \ = M_{[1,1]}M_{[1,2]}M_{[2]} \ .$$

If $v_1$ is not an eigenvector of $M_{[1,1]}$, then $v_1$ and $v_2$ are linearly independent, and in three-dimensional space, the vector product of them $v_3 = v_1\times v_2$ defines a third nonzero covariant vector. With these three linearly independent vectors, the orthogonal transformation can be uniquely determined. In the two-dimensional case, only one nonzero covariant vector is necessary to determine the orientation of the shape.

### 5.2 Affine case

The determination of an intrinsic affine coordinate system differs from the Euclidean case. In the first place, although there are only $2^n$ orthogonal matrices which diagonalize a symmetric $n\times n$ matrix, the number of nonsingular matrices which diagonalize the same symmetric matrix is infinite. However, if two nonsingular matrices transform a positive definite matrix into the identity matrix, they are related by an orthogonal transformation.

**Lemma 4** *Let $M$ be a symmetric positive definite $n\times n$ matrix. Then,*

1. *The inverse of the lower triangular Cholesky decomposition of $M$ is the unique lower triangular matrix $L$, with positive diagonal elements, such that $LML^t = I$.*
2. *If $A$ and $B$ are two $n\times n$ matrices such that $AMA^t = BMB^t = I$, then $AB^{-1}$ is an orthogonal matrix. In particular, for every $n\times n$ matrix $A$ such that $AMA^t = I$, there exists a unique orthogonal matrix $Q$ such that $A = QL$.*

For the proof of 1 see Golub [Golub and Van Loan 1983], and for 2 just note that

$$I \ = \ AMA^t \ = \ A\left(B^{-1}B^{-t}\right)A^t \ = \ \left(AB^{-1}\right)\left(AB^{-1}\right)^t \ ,$$

or equivalently, $Q = AB^{-1}$ is orthogonal.

Now, let $M$ be any $n\times n$ covariant matrix of moments, such as $M_{[1,1]}$, or one of the matrices in (8). Let $L$ be the triangular matrix of the Lemma, and let us consider the coordinate transformation $x' = Lx$ defined by this matrix. Then, the corresponding covariant matrix $M'$ in the new coordinate system is the identity matrix, because $M' = LML^t = I$. In order to determine a canonical affine transformation, we still need to uniquely specify a canonical orthogonal matrix, because, for every orthogonal matrix $Q$, if $A = QL$, and $x'' = Qx' = Ax$ then, we also have $M'' = AMA^t = QQ^t = I$. After the coordinate transformation defined by $L$, we are in the Euclidean case, but since $M' = I$ has all the eigenvalues repeated, we cannot use this matrix to determine an orientation, and we have to consider *a second covariant matrix* $N$, with nonrepeated eigenvalues, for the determination of the rotation part of $A$. This orthogonal matrix is, as in the Euclidean case, one of the $2^n$ orthogonal matrices which diagonalize $N' = LNL^t$ leaving the eigenvalues in decreasing order.

In the applications we will take $M = M_{[1,1]}$, and

$$N \;=\; M'_{[1,k]}M'_{[k,1]} \;=\; [LM_{[1,k]}L^t_{[k]}][L_{[k]}M_{[k,1]}L^t]\;,$$

for the smallest value of $k = 2,3,\ldots$ for which $N$ has well separated eigenvalues. Finally, in the two and three-dimensional cases, we will use the covariant vectors

$$v_1 \;=\; M'_{[1,2]}M'_{[2]}$$

and

$$v_2 \;=\; M'_{[1,1]}v_1$$

to disambiguate among the $2^n$ candidate orthogonal transformations.

## 6 Appropriate regions for which to compute invariants

The conceptually simplest way to use invariants is to compute them for a curve in 2D that represents an entire object, or a surface in 3D that represents an entire object. Though this will often be appropriate, difficulties that can arise are the following:

1. The preceding requires entire object segmentation which may not be possible or easy if it is not model based.
2. The object may be partially occluded, in which case moment invariants for the data and for the appropriate stored model will be different, because they will not be for the same boundary regions of the object in the data and the object in the stored data base.
3. In the case of 3D data, it is usually impractical to collect data over the entire object surface. More generally, range data will be collected from one direction or from a few directions, but self occlusion will prevent a sizable portion of the object surface from being sensed.

Hence, an alternative to object recognition by computing moment invariants for an entire object is to compute moment invariants for each of a few regions for each object. We call these regions *interest regions*. They should be such that they are easily identified and should provide good object discriminatory power. Then, for each object to be recognized, a vector of moment invariants is stored in a data base. A subset of the components of such a vector constitutes the moment invariants used for one interest region. Hence, for object recognition, interest regions are found in the data set in order to do preliminary object recognition. Thus interest regions serve two purposes. The first is that even though partial object occlusion may occur, one or a few of the interest regions for an object will be observed. The second is that the interest regions are selected to be sufficiently distinctive that the object will be machine recognizable at least at some modest but useful level of accuracy if one or more interest regions is observable in the data. What is required here is that the object sensed in the data be recognized to the extent that its possible object classes are reduced to a relatively small number at modest computational cost. Distinguishing which of these small number of objects or object classes is correct can then be done with more extensive processing.

Key to this approach is using a computationally cheap interest region finder that finds interest regions irrespective of the object they belong to. When the invariance is for Euclidean transformations, one approach to this problem is to find regions, that are discs in the plane or spheres in three-space, of one or a few sizes such that the enclosed data is an interest region of sufficient complexity to narrow down the possible objects containing the interest region to a few. Hence, the trick is to have a computationally inexpensive way of measuring the complexity of the region, and of locating the region center exactly. Assuming that the data in most regions is well represented by a first or second degree algebraic curve or surface, one approach to the complexity problem is to find regions that are well approximated by a third or fourth degree degree algebraic curve or surface. If there are few of these, they are apt to be useful for object recognition. Alternatively, if the number of discs (spheres) in which the data is not well approximated by first or second degree polynomials is small, then these regions could be treated as potential interest regions without further assessing the complexities of their algebraic representations. Finding these regions might proceed as follows. Cover the data set with a regular array of discs (spheres) of one or a few sizes. Fit straight lines (planes) to the data in all the regions. If a region is not well fit by a straight line, then fit a conic (quadric), and mark those regions that are not well fit by by a conic. These are interest regions.

Two remaining questions to be addressed are first, how are moments computed for open curves or surfaces, and second how are the regions located exactly. For Euclidean invariance, the first problem can be handled in either of two ways. It can be handled as in preceding sections, where the moments can be computed by doing simple spline curve (or surface) approximation to curve data in 2D (or range data in 3D), and then computing the moments by integration using uniform measure along the curve spline (surface spline). Alternatively, the problem can be handled by computing area (volume) moments as illustrated in figure 1. Let A and B denote the points where the curve intersects the disc defining the interest region. Draw the straight line $\overline{\mathrm{AB}}$ connecting points A and B. The area within the disc that is bounded by the curve and by the line $\overline{\mathrm{AB}}$ is the area used in the moment computation.

It is possible for a curve to intersect a disc in more than two points. However, we can still deal with sufficiently general curves by limiting consideration to curve segments intersecting discs in only two points (larger numbers of intersections can be dealt with if necessary).

For affine invariance, it is no longer correct to compute moments as curve (surface) integrals because the measure along the curve (surface) is no longer uniform. Then the area (volume) computation of moments must be made.

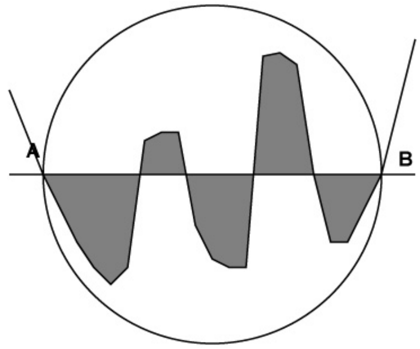


Figure 1: Area used to compute moments of an open interest region.

Finally, how are the discs located precisely ? A number of different functions of disc location can be chosen for the purpose. One is to maximize the area defined by the curve in the disc and the associated line $\overline{AB}$. Another is to maximize the determinant of the second moment matrix $M_{[1,1]}$ for the region. The determinant is a measure of the square of the volume occupied by the data.

Finding interest regions for the case where a curve has undergone an apriori unknown affine transformation is somehow more complicated, because a fixed region shape such as a disc cannot be used. We are presently determining possible methods. Of course, an approach to curve segmentation that is in widespread usefor a variety of purposes is to segment a curve at points of discontinuity of the curve, tangent or normal. However, we are interested in more general situations such as disconnected curves, and curves without tangent or normal discontinuities.

# 7 Examples and implementation details

## 7.1 Affine moment invariants of 2D shapes

A simple example of classification of 2D shapes based on affine moment invariants is illustrated in figures 2 and 3, and tables 4 and 5.

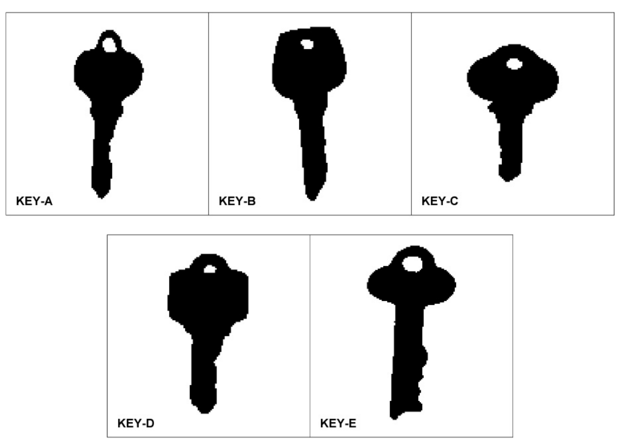


Figure 2: Example of affine moment invariants : model images.

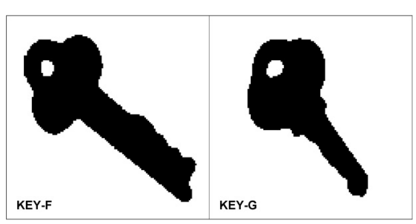


Figure 3: Example of affine moment invariants : test images.

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| KEY-A | 260 | 391 | 536 | 1053 | 1605 | 139 | 413 | 1596 |
| KEY-B | 261 | 367 | 594 | 937 | 1533 | 155 | 347 | 1532 |
| KEY-C | 209 | 405 | 462 | 1106 | 1574 | 97 | 467 | 1571 |
| KEY-D | 190 | 311 | 528 | 992 | 1521 | 100 | 315 | 1521 |
| KEY-E | 267 | 310 | 720 | 816 | 1706 | 214 | 264 | 1621 |
| KEY-F | 218 | 241 | 708 | 802 | 1653 | 165 | 202 | 1581 |
| KEY-G | 223 | 415 | 468 | 1113 | 1583 | 104 | 477 | 1582 |

Figure 4: Example of affine moment invariants : vectors of affine invariants corresponding to the model and test images of figures 2 and 3.

| | KEY-F | KEY-G |
|---|---|---|
| KEY-A | 406 | 127 |
| KEY-B | 295 | 274 |
| KEY-C | 511 | **27** |
| KEY-D | 335 | 253 |
| KEY-E | **135** | 489 |

Figure 5: Example of affine moment invariants : distances among vectors of affine invariants of table 4. KEY-F is closer to KEY-E, and KEY-G is closer to KEY-C.

The eight numbers associated with each model and test image in table 4 are affine moment invariants. They are The first two are the eigenvalues of the symmetric $2\times 2$ matrix $M'_{[1,2]}M'_{[2,1]}$, the second three are the eigenvalues of the $3\times 3$ matrix $M'_{[2,2]}$, the following two invariants are the eigenvalues of the $2\times 2$ matrix $M'_{[1,2]}M'_{[2,2]}M'_{[2,1]}$, and the last invariant is $M'_{[0,2]}M'_{[2,2]}M'_{[2,0]}$. The values are displayed after being multiplied by a suitable constant and rounded to the closest integer value. These centered moments are computed not with respect to the original coordinate system, but with respect to the coordinate system defined by $x' = Lx$, where $L$ is a $2\times 2$ lower triangular matrix such that $LM_{[1,1]}L^t = I$, and $M_{[1,1]}$ is the $2\times 2$ matrix of moments with respect to the original coordinate system.

In order to show how simple these computations are, we now describe how these five affine moment invariants are evaluated. Let $\{p_1,\ldots,p_q\}$ be the set of pixels in one of the black areas of figure 2 or 3. The first operation is to compute its center

$$\begin{aligned} \bar{x}_1 &= \tfrac{1}{q}\textstyle\sum_{i=1}^{q} p_{i1} \\ \bar{x}_2 &= \tfrac{1}{q}\textstyle\sum_{i=1}^{q} p_{i2} \end{aligned}$$

where $p_{i1}$ and $p_{i2}$ are the two coordinates of the point $p_i$. Then we compute the centered second degree moments

$$\begin{aligned} M_{(2,0)} &= \tfrac{1}{q}\textstyle\sum_{i=1}^{q}(p_{i1}-\bar{x}_1)^2 \\ M_{(1,1)} &= \tfrac{1}{q}\textstyle\sum_{i=1}^{q}(p_{i1}-\bar{x}_1)(p_{i2}-\bar{x}_2) \\ M_{(0,2)} &= \tfrac{1}{q}\textstyle\sum_{i=1}^{q}(p_{i2}-\bar{x}_2)^2 \end{aligned}$$

which we rearrange as a $2\times 2$ matrix

$$M_{[1,1]} = \begin{pmatrix} M_{(2,0)} & M_{(1,1)} \\ M_{(1,1)} & M_{(0,2)} \end{pmatrix} .$$

The third step is to find the lower triangular matrix $L$
such that $M'_{[1,1]} = LM_{[1,1]}L^t = I$. We compute it in two steps; we first find the lower triangular matrix $L$ such that $LL^t = M_{[1,1]}$, the Cholesky decomposition of $M_{[1,1]}$

$$L = \begin{pmatrix} L_{11} & 0 \\ L_{21} & L_{22} \end{pmatrix} \qquad \text{with} \qquad \begin{cases} L_{11} = \left(M_{(2,0)}\right)^{1/2} \\ L_{21} = M_{(1,1)}/L_{11} \\ L_{22} = \left(M_{(0,2)} - L_{21}^2\right)^{1/2} \end{cases} ,$$

and then we invert it in place

$$\begin{cases} L_{11} = 1/L_{11} \\ L_{21} = -L_{21}L_{11}/L_{22} \\ L_{22} = 1/L_{22} \end{cases}$$

At this point we compute moments of degree three, four, and eventually higher degree

$$M'_{(i,j)} = \frac{1}{q}\sum_{i=1}^{q}[L_{11}(p_{i1}-\bar{x}_1)]^i[L_{21}(p_{i1}-\bar{x}_1)+L_{22}(p_{i2}-\bar{x}_2)]^j$$

for $i+j>2$. The first two affine invariants are the two eigenvalues of the $2\times 2$ symmetric positive definite matrix $M'_{[1,2]}M'_{[2,1]} = M'_{[1,2]}M'^t_{[1,2]}$, where

$$M'_{[1,2]} = M'^t_{[2,1]} = \begin{pmatrix} \frac{1}{\sqrt{2}}M'_{(3,0)} & M'_{(2,1)} & \frac{1}{\sqrt{2}}M'_{(1,2)} \\ \frac{1}{\sqrt{2}}M'_{(2,1)} & M'_{(1,2)} & \frac{1}{\sqrt{2}}M'_{(0,3)} \end{pmatrix} .$$

The second two affine moment invariants of table 4 are the eigenvalues of the $3\times3$, symmetric nonnegative definite matrix

$$M'_{[2,2]} = \begin{pmatrix} \frac{1}{2}M'_{(4,0)} & \frac{1}{\sqrt{2}}M'_{(3,1)} & \frac{1}{2}M'_{(2,2)} \\ \frac{1}{\sqrt{2}}M'_{(3,1)} & M'_{(2,2)} & \frac{1}{\sqrt{2}}M'_{(1,3)} \\ \frac{1}{2}M'_{(2,2)} & \frac{1}{\sqrt{2}}M'_{(1,3)} & \frac{1}{2}M'_{(0,4)} \end{pmatrix} .$$

The following two affine invariants are the eigenvalues of the product of the previous matrices $M'_{[1,2]}M'_{[2,2]}M'_{[2,1]}$, and the last affine moment invariant is just

$$M'_{[0,2]}M'_{[2,2]}M'_{[2,0]} = \frac{1}{4}M'_{(4,0)} + \frac{1}{4}M'_{(0,4)} ,$$

because, since $M'_{[1,1]} = I$, we have

$$M'_{[0,2]} = M'_{[2,0]} = \begin{pmatrix} \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \end{pmatrix} .$$

## 7.2 Euclidean moment invariants of 3D surface regions

An example of computation of moment invariants from range maps of 3D surface regions of simple geometrical shapes is illustrated in figure 6 and table 7, with table 8 showing the Euclidean distances among the vectors of invariants of table 7. Then two other examples are shown where the data sets are subsets of range maps of faces. The result of computing the intrinsic Euclidean coordinate frames of some of these regions is shown in figure 12. A second example is shown in figure 9 and table 10, with the distances among the vectors of invariants in table 11. In this second example three noses are compared with three eyes. The distances between invariant vectors is very small between members of each group, and large between members of different groups. A third example is shown in figure 13 and table 14, with the distances among the vectors of invariants in table 15. In this example corresponding regions of range maps of the same object, but taken from different orientations are compared. In the three cases the source images belong to the NRCC three-dimensional image data files[Rioux and Cournoyer 1988]. Also, in the three cases the data sets are the data points contained inside a sphere of a fixed radius (20 pixels) centered at one of the data points. Centering of the spheres was done by an observer, but in practice it would be done automatically by methods such as those described in section 6. The contours of these data sets, shown in black in the figures, correspond to the intersections of these spheres with the surfaces determined by the range images.

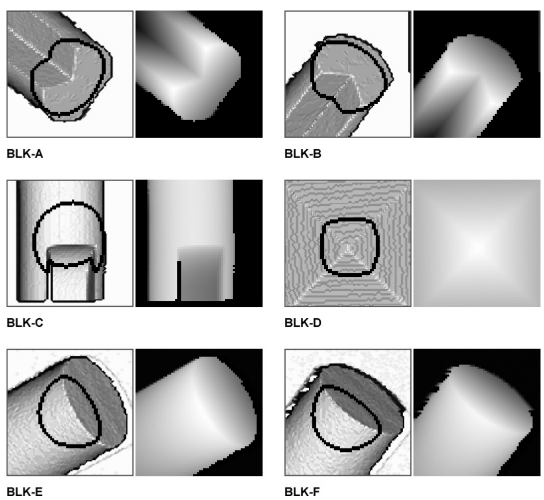


Figure 6: Example of 3D Euclidean moment invariants : regions of simple geometric shapes.

All the derivations of moment invariants are based on the assumption of an *exact* correspondence between two measures which describe the data in two different coordinate systems. This assumption is violated in this case, where the data set is just a finite set of samples of the surface of the object. These points are not uniformly distributed along the surface; their distribution depends on the relative orientation of

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| BLK-A | 20 | 90 | 117 | 214 | 247 | 523 | 652 | 1202 | 1570 |
| BLK-B | 15 | 79 | 100 | 222 | 243 | 532 | 612 | 1230 | 1482 |
| BLK-C | 20 | 25 | 65 | 102 | 249 | 362 | 462 | 1144 | 1397 |
| BLK-D | 10 | 50 | 69 | 71 | 219 | 310 | 324 | 854 | 860 |
| BLK-E | 8 | 42 | 66 | 92 | 274 | 300 | 381 | 849 | 1235 |
| BLK-F | 8 | 41 | 63 | 89 | 269 | 292 | 374 | 827 | 1221 |

Figure 7: Example of 3D Euclidean moment invariants : vectors of invariants corresponding to the surface regions of figure 6.

| | BLK-A | BLK-B | BLK-C | BLK-D | BLK-E | BLK-F |
|---|---|---|---|---|---|---|
| BLK-A | 0 | 104 | 338 | 896 | 617 | 644 |
| BLK-B | 104 | 0 | 291 | 828 | 578 | 605 |
| BLK-C | 338 | 291 | 0 | 630 | 353 | 381 |
| BLK-D | 896 | 828 | 630 | 0 | 384 | 370 |
| BLK-E | 617 | 578 | 353 | 384 | 0 | 29 |
| BLK-F | 644 | 605 | 381 | 370 | 29 | 0 |

Figure 8: Example of 3D Euclidean moment invariants : distances among vectors of invariants. BLK-A is close to BLK-B, and BLK-E is close to BLK-F.

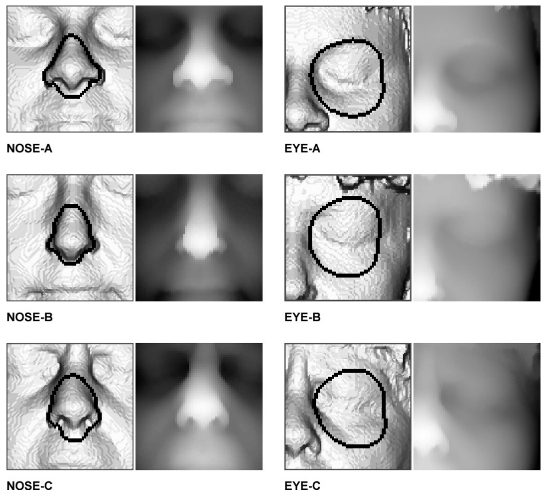


Figure 9: Example of 3D Euclidean moment invariants : surface regions of faces.

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| NOSE-A | 21 | 57 | 109 | 113 | 168 | 246 | 347 | 676 | 801 |
| NOSE-B | 9 | 66 | 105 | 140 | 168 | 220 | 357 | 581 | 742 |
| NOSE-C | 20 | 58 | 98 | 128 | 203 | 250 | 353 | 671 | 859 |
| EYE-A | 9 | 40 | 68 | 92 | 254 | 489 | 529 | 1427 | 1483 |
| EYE-B | 12 | 23 | 42 | 81 | 241 | 494 | 506 | 1376 | 1497 |
| EYE-C | 15 | 32 | 44 | 118 | 238 | 501 | 526 | 1404 | 1498 |

Figure 10: Example of 3D Euclidean moment invariants : vectors of invariants corresponding to the surface regions of figure 9.

| | NOSE-A | NOSE-B | NOSE-C | EYE-A | EYE-B | EYE-C |
|---|---|---|---|---|---|---|
| NOSE-A | 0 | 119 | 70 | 1064 | 036 | 1060 |
| NOSE-B | 119 | 0 | 155 | 1173 | 146 | 1168 |
| NOSE-C | 70 | 155 | 0 | 1027 | 998 | 1022 |
| EYE-A | 1064 | 1173 | 1027 | 0 | 67 | 50 |
| EYE-B | 1036 | 1146 | 998 | 67 | 0 | 52 |
| EYE-C | 1060 | 1168 | 1022 | 50 | 52 | 0 |

Figure 11: Example of 3D Euclidean moment invariants : distances among the vectors of invariants of table 10. Difference between NOSE's and EYE's can be observed in this examples.

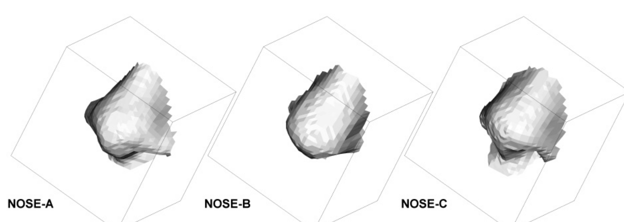


Figure 12: Some of the surface regions of figure 9, an their canonical Euclidean frames of reference.

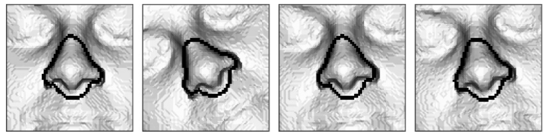


Figure 13: Example of Euclidean moment invariance variation: surface regions

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| NOSE-D | 19 | 59 | 111 | 115 | 172 | 241 | 350 | 675 | 780 |
| NOSE-E | 20 | 56 | 103 | 117 | 171 | 246 | 344 | 683 | 774 |
| NOSE-F | 20 | 60 | 113 | 116 | 171 | 244 | 349 | 688 | 795 |
| NOSE-G | 21 | 58 | 109 | 116 | 170 | 248 | 347 | 696 | 787 |

Figure 14: Examplele of Euclidean moment invariance variation : vectors of invariants corresponding to the surface regions of figure 13.

the sensor with respect to the surface. In order to recover the invariance, the points are used to define a triangulation of the surface, and the moments are evaluated as the sum of the surfaces integrals over the triangles. With a triangulation we recover the uniform distribution with respect to surface area. For this we need explicit formulas, and algorithms, to evaluate moments on a 3D triangular planar patch, the convex hull of a set of three 3D points. These formulas can be derived based on the representations of nonsingular matrices introduced above, but due to lack of space we will address this problem in a future report.

The nine Euclidean invariants shown in the tables are the generalized eigenvalues of the pencil of block matrices

$$\begin{pmatrix} M_{[1,1]} & M_{[1,2]} \\ M_{[2,1]} & M_{[2,2]} \end{pmatrix} - \lambda \begin{pmatrix} N_{[1,1]} & N_{[1,2]} \\ N_{[2,1]} & N_{[2,2]} \end{pmatrix} ,$$

We have decided to use these invariants because there is a geometric interpretation for some of them. For example, the eigenvector corresponding to the minimum eigenvalue is, except for the constant term, the vector of coefficients of the second degree polynomial whose associated set of zeros, i.e., associated surface, approximately best fits the data in the least squares sense. This minimum generalized eigenvalue is approximately the sum of the squares of the perpendicular distances from the data points to the approximating algebraic surface [Taubin 1988a; Taubin 1988b; Taubin 1991a].

In this way we can use this same computation to search for interest regions, as described in section 6.

## 8 Related work on moment invariants

Several authors have considered moment based methods for object recognition and positioning, first for two-dimensional, and more recently for three-dimensional objects, spanning a period of almost thirty years [Hu 1962; Alt 1962; Udagawa et al. 1964; Smith and Wright 1971; Dudani et al. 1977; Dirilten and Newman 1977; Wong and Hall 1978; Maitra 1979; Sadjadi and Hall 1978; Sadjadi and Hall 1980; Teague 1980; Reddi 1981; Casasent et al. 1982; Faugeras and Hebert 1983; Boyce and Hossack 1983; Kanatani 1984a; Kanatani 1984b; Abu-Mostafa and Psaltis 1984; Abu-Mostafa and Psaltis 1985; Pinjo et al. 1985; Cygansky and Orr 1985; Lin et al. 1986; Cash and Hatamian 1987; Taylor and Reeves 1987; Zakaria et al. 1987; Lo and Don 1987; Hong and Tan 1987a; Hong and Tan 1987b; Teh and Chin 1988a; Teh and Chin 1988b; Faber and Stokely 1988; Taubin et al. 1989; Lo and Don 1989]. The first to introduce moment invariants in the Pattern Recognition literature, was Hu [Hu 1962]. He presented a theory of two-dimensional moment invariants for planar geometric figures based on the classic theory of algebraic invariants of binary forms. He derived complete systems of two-dimensional moment invariants under Euclidean transformations, expressing the rotations as multiplications by exponentials in the complex plane. He also included some affine moment invariants. For the second and third order moments he derived seven orthogonal invariants which are functionally equivalent to some of the invariants described in this paper.

Several other researchers used Hu's invariants for different purposes. Dudani, Breeding, and McGhee [Dudani et al. 1977] used the seven two-dimensional moment invariants of Hu for the identification of aircraft from their projected contours. Wong and Hall [Wong and Hall 1978] used the seven two-dimensional moment invariants of Hu for the matching of radar to optical images using a hierarchical search technique with the moment invariants as similarity measures. Maitra [Maitra 1979] modified Hu's seven orthogonal invariants to make them also invariant under scale and illumination changes. Sadjadi and Hall [Sadjadi and Hall 1978] studied numerical methods for the evaluation of the Hu's seven moment invariants. Later [Sadjadi and Hall 1980], they partially extend Hu's work to the three-dimensional case. Based

| | NOSE-D | NOSE-E | NOSE-F | NOSE-G |
|---|---|---|---|---|
| NOSE-D | 0 | 15 | 20 | 24 |
| NOSE-E | 15 | 0 | 24 | 20 |
| NOSE-F | 20 | 24 | 0 | 13 |
| NOSE-G | 24 | 20 | 13 | 0 |

Figure 15: Example of Euclidean moment invariance variation : distances among the vectors of invariants of table 14.

on the theory of algebraic forms, they develop certain orthogonal invariants of quadratic and cubic forms, which are particular cases of the methods described in this and previous chapters.

Several authors derived moment invariants under constrained families of affine transformations, and also considered moments with respect to other functions, not polynomials. Alt [Alt 1962] used moments as invariant features for the recognition of printed symbols, under a limited family of affine transformations. Teague [Teague 1980] introduced Zernike moments as features for the recognition of two-dimensional patterns, and established their relations with respect to the usual moments. Reddi [Reddi 1981] defined angular and radial moments, and established their relation with Hu's seven moment invariants. Casasent, Cheatham, and Fetterly [Casasent et al. 1982] described an optical system to compute intensity moments of two-dimensional images. Boyce and Hossack [Boyce and Hossack 1983] used Zernike moments for image reconstruction. Abu-Mostafa and Psaltis [Abu-Mostafa and Psaltis 1984] evaluated the two-dimensional moment invariants as features for pattern recognition in terms of discrimination power and noise tolerance. Later [Abu-Mostafa and Psaltis 1985], they considered a new normalization process for two-dimensional images based on complex moments. Cash and Hatamian [Cash and Hatamian 1987] used moments as invariant features for the recognition of characters in printed documents. Teh and Chin [Teh and Chin 1988a; Teh and Chin 1988b] compared different types of moments, regular moments, Legendre moments, Zernike moments, pseudo-Zernike moments, rotational moments, and complex moments, with respect to the representation and recognition of two-dimensional patterns.

Affine normalization of two-dimensional shapes is a subject treated by different authors as well. Udagawa, Toriwaki, and Sugino [Udagawa et al. 1964] defined a procedure for the normalization of two-dimensional patterns under affine transformations, capital letters in their examples, based on moments, and used the normalized moments as invariant features for recognition. Dirilten and Newman [Dirilten and Newman 1977] were concerned with the problems of recognition and positioning of patterns under affine transformations. They showed that there are infinitely many affine transformations which make the moments up to degree two of two patterns match, and two of them differ by an orthogonal transformation, following the same approach that we have followed, but they do not show a direct method to recover the unknown orthogonal transformation. They also derived certain orthogonal moment invariants by contracting indices of the symmetric moment tensors. These invariants can also be obtained with the methods described here. For simplicity, and because the treatment presented in the text was sufficient for our purposes, we have deliberately omitted to introduce tensors, and to mention the relation between symmetric tensors, forms and moments. Faber and Stokely [Faber and Stokely 1988] determined the affine transformation which relates two three-dimensional shapes by computing four pairs of covariant points using tensor-based techniques, and then solving the linear system which results from the pairing. These covariant points usually involve moments of degree up to five. They also used the method of the principal directions of the tensor of inertia, a covariant matrix of second degree moments, for recovering Euclidean transformations.

Hong and Tan [Hong and Tan 1987a; Hong and Tan 1987b] introduced the concept of moment curve of a set of points, as a tool for the affine normalization of planar shapes. The moment curve of a shape is an algebraic curve of degree two or three, with its coefficients functions of the second degree moments of a set of points. It is a circumference if and only if the matrix of second degree moments $M_{[1,1]}$ is a multiple of the identity matrix, and two shapes are equivalent with respect to affine transformations, if and only if their corresponding moment curves are equivalent with respect to orthogonal transformations. They propose as a dissimilarity function between two shapes, the minimum, over all the rotations, of an orthogonal dissimilarity function between the corresponding moment curves. The orthogonal dissimilarity function is based on heuristics, and involves rotating one curve to a finite number of angles, and comparing it with the other. Using the implicit equation of the moment curve, we could use the methods for curve positioning to improve their method, but it is less expensive to recover the affine transformation directly from the moments, as we have explained above.

The Euclidean and affine matching problems are also related to the motion estimation problem. Lin, Lee, and Huang [Lin et al. 1986] estimated the Euclidean transformation which transforms one set of points into a second one. They computed the translation part the difference between the centers of both sets, and the rotation part by diagonalizing the scatter matrices $M_{[1,1]}$ and $M'_{[1,1]}$, obtaining, as we did, $2^n$ candidate solutions, or $2^{n-1}$ if only proper orthogonal matrices are allowed. The method that they proposed for discriminating among these $2^{n-1}$ candidate transformations is not direct though, and has a complexity function of the number of points.

A few authors have worked out extensions of Hu's invariants to the three-dimensional case. Pinjo, Cyganski and Orr [Pinjo et al. 1985; Cygansky and Orr 1985] described moment based methods for the determination of the orientation of 3-D objects in 3-space either from 2-D projections or 3-D surface coordinate information. Their methods require the computation of moments up to degree five. Lo and Don [Lo and Don 1987; Lo and Don 1989] developed three-dimensional orthogonal moment invariants using complex moments and the irreducible decomposition of the representation of the orthogonal group defined by these moments. This approach produces invariants which are functionally equivalent to those produced by what we have called elsewhere the harmonic decomposition [Taubin 1991b]. They also determined $2^n$ candidate Euclidean transformations for matching two sets of points, by centering the moments and diagonalizing the matrix of second degree moments. They discriminate among these $2^n$ candidates by looking at third degree moments, as we do, obtaining a totally equivalent method for position estimation.

Finally, Bamieh and DeFigueiredo [Bamieh and deFigueiredo 1986] used affine moment invariants of planar poligonal regions for the recognition of 3D polyhedral objects from the projected contours of their faces onto the image plane. They derived affine moment invariants of planar shapes using tensor calculus, and presented an algorithm, based on Green's theorem, for the computation of moments of polygonal regions from the coordinates of the vertices.

## 9 Conclusions

By introducing the concept of covariant matrix, we have been able to define efficient algorithms for the computation of Euclidean and affine moment invariants, and for the Euclidean and affine normalization of 2D and 3D shapes. These invariants permit low computation matching of a subobject in arbitrary position in the data to a subobject stored in standard position in a data base. We also use these invariants to define an intrinsic coordinate system — a center, an orientation, and a stretching — that, among other things permits computing the relative position of a subobject in the data and a subobject in the data base. All these methods are based on well established and simple matrix computation techniques. Except for the computation of the moments themselves, the complexity of all these algorithms is polynomial in the number of moments involved, as opposed to most previously known algorithms, which solve just part of the problems covered with the methods described in this paper, and usually have complexity function of the number of points, require nonlinear optimization methods, or both. Finally, the implementation of object recognition and position estimation systems based on these methods is under way, and will be reported in the near future.

In section 3 we pointed out the one-to-one correspondence of moment invariants with algebraic invariants, i.e., invariants that are funcions of the coefficients of polynomials which define curves or surfaces. The question that then arises is what are the relative merits of moment and algebraic curve and surface invariants ? A definite answer remains to be determined. Two relative merits that are immediately apparent are the following :

1. A few invariants based on low order moments will often be adequate for recognition of complex curve or surface objects. The computation here is very low. Hence, moment invariants are computationally atractive. However, the computation of algebraic invariants first requires accurate fitting of a complex curve or surface to the data. This computation can be much greater than that required for the computation of the moments.
2. A polynomial curve or surface that represents an object can often be fit very accurately even if a sizable subset of the data along an object boundary is missing. The algebraic invariants are then still effective for object recognition and position estimation, whereas the moment invariants probably would not be useful because they would change due to the missing data.

## 10 Appendix : proofs

*Proof of Lemma 1 :* The *multinomial formula* is

$$\frac{1}{d!}(x_1+\cdots+x_n)^d \;=\; \sum_{|\alpha|=d}\frac{1}{\alpha!}x_1^{\alpha_1}\cdots x_n^{\alpha_n} \;=\; \sum_{|\alpha|=d}\frac{1}{\alpha!}x^{\alpha}\;.$$

Let $x$ and $y$ be two $n$-dimensional vectors, and let us consider the multinomial expansion of the $d$-th power of the inner product $y^t x$ , the polynomial of $2n$ variables

$$\begin{array}{rcl}\frac{1}{d!}(y^t x)^d &=& \frac{1}{d!}(y_1x_1+\cdots+y_nx_n)^d\\ &=& \sum_{|\alpha|=d}\frac{1}{\alpha!}(y_1x_1)^{\alpha_1}\cdots(y_nx_n)^{\alpha_n}\\ &=& \sum_{|\alpha|=d}\frac{1}{\alpha!}y^{\alpha}x^{\alpha}\;.\end{array}$$

This polynomial is homogeneous of degree $d$ in both $x$ and $y$ , and it is obviously invariant under simultaneous orthogonal transformations of the variables $x$-$y$ . In vector form,

$$\frac{1}{d!}(y^t x)^d \;=\; X_{[d]}(y)^t X_{[d]}(x)\;.$$

1-(a).) Let $A$ and $B$ be $n\times n$ nonsingular matrices. Then, the following expression

$$\begin{array}{rclcl}(AB)_{[d]}X_{[d]}(x) &=& X_{[d]}((AB)x) &=& X_{[d]}(A(Bx))\\ &=& A_{[d]}X_{[d]}(Bx) &=& A_{[d]}(B_{[d]}X_{[d]}(x))\\ &=& (A_{[d]}B_{[d]})X_{[d]}(x) && \end{array}$$

is a polynomial identity, and all the coefficients of the polynomials on the left side are identically to the corresponding coefficients of the polynomials on the right side, that is

$$(AB)_{[d]} \;=\; (A_{[d]}B_{[d]})\;.$$

1-(b).) Follows from the uniqueness of representation of a homogeneous polynomial as a linear combination of monomials (5).

1-(c).) From 1-(b).), the identity matrix is map to the identity matrix. Let $A$ be a $n\times n$ nonsingular matrix. Apply 1-(a).) with $B=A^{-1}$ to obtain

$$I \;=\; (AA^{-1})_{[d]} \;=\; A_{[d]}(A^{-1})_{[d]} \quad\Rightarrow\quad (A_{[d]})^{-1} \;=\; (A^{-1})_{[d]}\;.$$

2.) Let $A$ be a $n\times n$ nonsingular matrix Then, the following expression

$$\begin{array}{rcl}0 &=& \frac{1}{d!}[((Ay)^t x)^d-(y^t(A^t x))^d]\\ &=& X_{[d]}(Ay)^t X_{[d]}(x)-X_{[d]}(y)X_{[d]}(A^t x)\\ &=& X_{[d]}(y)^t\left((A_{[d]})^t-(A^t)_{[d]}\right)X_{[d]}(x)\end{array}$$

is a polynomial identity, and all the coefficients of the polynomial on the right side are identically zero, that is

$$(A^t)_{[d]} \;=\; (A_{[d]})^t \;.$$

If $A$ is symmetric, we have

$$(A_{[d]})^t \;=\; (A^t)_{[d]} \;=\; A_{[d]} \;.$$

If the matrix $A$ is symmetric positive definite, we can write $A = BB^t$, for certain nonsingular $n\times n$ matrix $B$. Then

$$A_{[d]} \;=\; (BB^t)_{[d]} \;=\; B_{[d]}B^t_{[d]}$$

and so $A_{[d]}$ is positive definite as well. If $A$ is orthogonal, we have

$$(A_{[d]})^{-1} \;=\; (A^{-1})_{[d]} \;=\; (A^t)_{[d]} \;=\; (A_{[d]})^t \;.$$

3.) If $\alpha$ and $\beta$ are two multiindices of size $d$, the $(\alpha,\beta)$-th element of the matrix $A_{[d]}$ is

$$\sqrt{\tfrac{1}{\alpha!\,\beta!}}\, D^{\beta}\left((Ax)^{\alpha}\right) ,$$

Where $D^{\beta}$ is the partial differential operator

$$D^{\beta} \;=\; \left(\frac{\partial}{\partial x_1}\right)^{\beta_1} \cdots \left(\frac{\partial}{\partial x_n}\right)^{\beta_n} .$$

If $\beta$ follows $\alpha$ in the lexicographical order, then, for certain $1 < k < n$ we have

$$\alpha_1 = \beta_1, \ldots, \alpha_{k-1} = \beta_{k-1}, \alpha_k > \beta_k \;,$$

and so

$$\alpha_{k+1} + \cdots + \alpha_n \;<\; \beta_{k+1} + \cdots + \beta_n \;.$$

Since the matrix $A$ is lower triangular, the degree of

$$(Ax)^{\alpha} \;=\; \prod_{i=1}^{n} \left( \sum_{j=1}^{i} a_{ij}\, x_j \right)^{\alpha_i}$$

as a polynomial in $x_{k+1}, \ldots, x_n$ with coefficients polynomials in
$x_1, \ldots, x_k$ is clearly not greater than $\alpha_{k+1} + \cdots + \alpha_n$, and so

$$\left(\frac{\partial}{\partial x_{k+1}}\right)^{\beta_{k+1}} \cdots \left(\frac{\partial}{\partial x_n}\right)^{\beta_n} \left((Ax)^{\alpha}\right) \;=\; 0 \,.$$

It follows that $D^{\beta}((Ax)^{\alpha}) = 0$, and the matrix $A_{[d]}$ is lower triangular.

4.) For every matrix $A$, there exist an orthogonal matrix $Q$, and a lower triangular matrix $L$ such that $A = LQ$. Since the map $A \mapsto A_{[d]}$ is a homomorphism, we have $A_{[d]} = L_{[d]}Q_{[d]}$, where $L_{[d]}$ is lower triangular and $Q_{[d]}$ is orthogonal, i.e., the decomposition is preserved. Since $|A_{[d]}| = |L_{[d]}|$, without loss of generality we will assume that $A$ is lower triangular itself.

Now note that for every $1 \le k \le n$ the variable $x_k$ appears only in the last term of the product

$$\prod_{i=1}^{k} \left( \sum_{j=1}^{i} a_{ij}\, x_j \right)^{\alpha_i} \quad ,$$

and so

$$\left(\frac{\partial}{\partial x_k}\right)^{\alpha_k} \left( \prod_{i=1}^{k} \left( \sum_{j=1}^{i} a_{ij}\, x_j \right)^{\alpha_i} \right) \;=\; \left( \prod_{i=1}^{k-1} \left( \sum_{j=1}^{i} a_{ij}\, x_j \right)^{\alpha_i} \right) \; \alpha_k!\, a_{kk}^{\alpha_k} \,.$$

By induction in $k = n, n-1, \ldots, 1$, it follows that the $\alpha$-th element of the diagonal of $A_{[d]}$ is

$$\tfrac{1}{\alpha!} D^{\alpha}\left((Ax)^{\alpha}\right) \;=\; a_{11}^{\alpha_1} \cdots a_{nn}^{\alpha_n} \;=\; a^{\alpha}$$

Since $A$ is triangular, $|A| = a_{11} \cdots a_{nn}$, and we have

$$|A_{[d]}| \;=\; \prod_{|\alpha|=d} a^{\alpha} \;=\; a^{\gamma} \,,$$

where $\gamma = \sum_{|\alpha|=d} \alpha$ . By symmetry, all the components of the multiindex $\gamma$ are equal, and so, for every $1 \le i \le n$

$$\gamma_i = \sum_{|\alpha|=d} \alpha_i = \frac{1}{n} \sum_{i=1}^{n} \sum_{|\alpha|=d} \alpha_i = \sum_{|\alpha|=d} = \frac{d}{n} \binom{n+d-1}{n-1} = \binom{n+d-1}{n} = m \,.$$

Finally

$$|A_{[d]}| \;=\; \left( \prod_{i=1}^{n} a_{ii} \right)^m \;=\; |A|^m \,.$$

□

*Proof of Lemma 2 :* Since $M_{[d]} = M_{[d,0]}$ we only need to prove the second part. First note that

$$X_{[j,k]}(Ax) \;=\; X_{[j]}(Ax) X^t_{[k]}(Ax) \;=\; A_{[j]} X_{[j,k]}(x) A^t_{[k]} \,.$$

In the case of a continuous data set

$$\begin{aligned} M'_{[k,j]} &= \frac{1}{|A||\mu|} \int X_{[k,j]}(A(x-\overline{\mu}))\,|A|\,d\mu(x) \\ &= \frac{1}{|\mu|} \int A_{[j]} X_{[k,j]}(x-\overline{\mu}) A_{[k]}\, d\mu(x) \quad = \quad A_{[j]} M_{[j,k]} A^t_{[k]} \,, \end{aligned}$$

by the well known change of variables formula. In the discrete case, the proof is the same, but without the $|A|$ .

□

*Proof of Lemma 3 :*

$$\begin{aligned} \Phi'^t_{[d]} X_{[d]}(x') &= \phi'(x') &&= \phi(A^{-1}x') \\ &= \Phi^t_{[d]} X_{[d]}(A^{-1}x') &&= \Phi^t_{[d]} \left[ A^{-1}_{[d]} X_{[d]}(x') \right] \\ &= \left[ A^{-t}_{[d]} \Phi_{[d]} \right]^t X_{[d]}(x') \end{aligned}$$

is a polynomial identity in $x'$ , and so, the coefficient vectors are equal

$$\Phi'_{[d]} \;=\; A^{-t}_{[d]} \Phi_{[d]} \,.$$

□

## References


ABU-MOSTAFA, Y., AND PSALTIS, D. 1984. Recognitive aspects of moment invariants. *IEEE Transactions on Pattern Analysis and Machine Intelligence 6*, 698–706.

ABU-MOSTAFA, Y., AND PSALTIS, D. 1985. Image Normalization by Complex Moments. *IEEE Transactions on Pattern Analysis and Machine Intelligence 7*, 46–55.

ACKERMAN, M. 1978. *Hilbert's Invariant Theory Papers*, vol. VIII of *Lie Groups: History, Frontiers and Applications*. Math Sci Press, Brookline, Massachusets.

ALT, F. 1962. Digital Pattern Recognition by Moments. *The Journal of the Association for Computer Machinery 9*, 240.

BAMIEH, B., AND DEFIGUEIREDO, R. 1986. A General Moment Invariants/Attributed-Graph Method for Three-Dimensional Object Recognition From a Single Image. *IEEE Transactions on Robotics And Automation 2*, 31–41.

BOOLE, G. 1841–1843. Exposition of a general theory of linear transformations. *Cambridge Mathematical Journal 3*, 1–20.

BOYCE, J., AND HOSSACK, W. 1983. Moment invariants for pattern recognition. *Pattern Recognition Letters 1*, 451–456.

CASASENT, D., CHEATHAM, L., AND FETTERLY, D. 1982. Optical system to compute intensity moments: Design. *Applied Optics 21*, 3292–3298.

CASH, G., AND HATAMIAN, M. 1987. Optical character recognition by the method of moments. *Computer Vision, Graphics, and Image Processing 39*, 291–310.

CAYLEY, A. 1889–1897. *The collected mathematical papers of Arthur Cayley*. Cambridge University Press, Cambridge.

CLEBSH, A. 1872. *Theorie der Binären Algebraischen Formen*. B.G. Teubner, Leipzig.

CYGANSKY, D., AND ORR, J. 1985. Applications of tensor theory to object recognition and orientation determination. *IEEE Transactions on Pattern Analysis and Machine Intelligence 7*, 662–673.

DICKSON, L. 1914. *Algebraic Invariants*, 1st. ed., vol. 14 of *Mathematical Monographs*. John Wiley & Sons Inc., New York.

DIRILTEN, H., AND NEWMAN, T. 1977. Pattern matching under affine transformations. *IEEE Transactions on Computers 26*, 314–317.

DUDANI, S., BREEDING, K., AND MCGHEE, R. 1977. Aircraft identification by moment invariants. *IEEE Transactions on Computers 26*, 39–45.

ELLIOT, E. 1913. *An Introduction to the Algebra of Quantics*, 2nd. ed. Oxford University Press, Oxford, UK.

FABER, T., AND STOKELY, E. 1988. Orientation of 3-D structures in medical images. *IEEE Transactions on Pattern Analysis and Machine Intelligence 10*, 626–633.

FAUGERAS, O., AND HEBERT, M. 1983. A 3D recognition and positioning algorithm using geometrical matching between primitive surfaces. In *Proceedings, 8th. International Joint Conference on Artificial Intelligence (IJCAI)*.

GOLUB, G., AND VAN LOAN, C. 1983. *Matrix Computations*. John Hopkins University Press.

GORDAN, P., AND KERSCHENSTEINER, G. 1887. *Vorlesungen über Invariantentheorie*. Liepzig.

GRACE, J., AND YOUNG, A. 1903. *The Algebra of Invariants*. Cambridge University Press.

GUREVICH, G. 1964. *Foundations of the Theory of Algebraic Invariants*. P. Noordhoff Ltd., Groningen, The Netherlands.

HILBERT, D. 1890. Über die Theorie der Algebraischen Formen. *Math. Ann. 36*, 473.

HILBERT, D. 1893. Über die vollen Invariantensysteme. *Math. Ann. 39*, 313.

HONG, J., AND TAN, X. 1987. A new approach to point pattern matching. Robotics Report 132, Courant Institute of Mathematical Sciences, New York University, New York, Dec.

HONG, J., AND TAN, X. 1987. The similarity between shapes under affine transformation. Robotics Report 133, Courant Institute of Mathematical Sciences, New York University, New York, December.

HU, M. 1962. Visual pattern recognition by moment invariants. *IRE Transactions on Information Theory 8*, 179–187.

KANATANI, K. 1984. Distribution of directional data and fabric tensors. *International Journal of Engineering Science 22*, 149–164.

KANATANI, K. 1984. Stereological determination of structural anisotropy. *International Journal of Engineering Science 22*, 531–546.

LIN, Z., LEE, H., AND HUANG, T. 1986. Finding 3-d point correspondences in motion estimation. In *Proceedings, Eight International Conference on Pattern Recognition*, 303–305.

LO, C., AND DON, H. 1987. Object identification and positioning via 3-d moments: a group-theoretic approach. Laboratory for Image Analysis and Graphics 87-06, State University of New York, Stony Brook, Oct.

LO, C., AND DON, H. 1989. 3-D moment forms: Their construction and application to object identification and positioning. *IEEE Transactions on Pattern Analysis and Machine Intelligence 11*, 1053–1064.

MAITRA, S. 1979. Moment invariants. *Proceedings of the IEEE 67*, 697–699.

MUMFORD, D. 1965. *Geometric Invariant Theory*, vol. 34 of *Ergebnisse der Mathematik und ihrer Grenzgebiete*. Springer-Verlag, New York.

PINJO, Z., CYGANSKI, D., AND ORR, J. 1985. Determination of 3-d object orientation from projections. *Pattern Recognition Letters 3*, 351–356.

REDDI, S. 1981. Radial and angular moment invariants for image identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence 3*, 240–242.

RIOUX, M., AND COURNOYER, L. 1988. The NRCC Three-dimensional Image Data Files. CNRC 29077, National Research Council of Canada, June.

ROTA, G., AND STURMFELS, B. 1989. *Invariant Theory and Tableaux*. No. 19 in The IMA Volumes in Mathematics and Its Applications. Springer-Verlag, ch. Introduction to Invariant Theory in Superalgebras, 1–35.

SADJADI, F., AND HALL, E. 1978. Numerical computations of moment invariants for scene analysis. In *Proceedings, IEEE Conference on Pattern Recognition and Image Processing*, 181–187.

SADJADI, F., AND HALL, E. 1980. Three-dimensional moment invariants. *IEEE Transactions on Pattern Analysis and Machine Intelligence 2*, 127–136.

SALMON, G. 1866. *Modern Higher Algebra*. Hodges, Smith and Co., Dublin, Ireland.

SMITH, F., AND WRIGHT, M. 1971. Automatic Ship Photo Interpretation by the Method of Moments. *IEEE Transactions on Computers 20*, 1089–1095.

SPRINGER, T. 1977. *Invariant Theory*, vol. 585 of *Lecture Notes in Mathematics*. Springer-Verlag, New York.

SYLVESTER, J. 1904–1912. *Collected Mathematical Papers*. Cambridge University Press, Cambridge, UK.

TAUBIN, G., AND COOPER, D. 1990. Recognition and positioning of 3d piecewise algebraic objects using euclidean invariants. In *Proceedings, Workshop on the Integration of Numerical and Symbolic Computing Methods*, Academic Press, Saratoga Springs, New York.

TAUBIN, G., BOLLE, R., AND COOPER, D. 1989. Representing and comparing shapes using shape polynomials. In *Proceedings, IEEE Conference on Computer Vision and Pattern Recognition*.

TAUBIN, G. 1988. Algebraic nonplanar curve and surface estimation in 3-space with applications to position estimation. LEMS 43, Brown University, Feb.

TAUBIN, G. 1988. Nonplanar Curve and Surface Estimation in 3-Space. In *Proceedings, IEEE Conference on Robotics and Automation*, vol. 1, 644–645.

TAUBIN, G. 1991. Estimation of planar curves, surfaces and nonplanar space curves defined by implicit equations, with applications to edge and range image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence 13*, 11 (Nov.), 1115–1138.

TAUBIN, G. 1991. *Recognition and Positioning of Rigid Object Using Algebraic and Moment Invariants*. PhD thesis, Brown University.

TAYLOR, R., AND REEVES, A. 1987. Three-dimensional image transformations in moment space. In *Proceedings of the IEEE Computer Society Workshop on Computer Vision*, 366–368.

TEAGUE, M. 1980. Image analysis via the general theory of moments. *Journal of the Optical Society of America 70*, 920–930.

TEH, C., AND CHIN, R. 1988. On image analysis by the methods of moments. In *Proceedings, IEEE Conference on Computer Vision and Pattern Recognition*, 556–561.

TEH, C., AND CHIN, R. 1988. On image analysis by the methods of moments. *IEEE Transactions on Pattern Analysis and Machine Intelligence 10*, 496–5113.

UDAGAWA, K., TORIWAKI, J., AND SUGINO, K. 1964. Normalization and recognition of two-dimensional patterns with linear distortion by moments. *Electronics and Communications in Japan 47*, 34–46.

WEYL, H. 1939. *The Classical Groups*. Princeton University Press, Princeton, New Jersey.

WHITE, N. 1989. *Invariant Theory and Tableaux*, vol. 19 of *The IMA Volumes in Mathematics and Its Applications*. Springer-Verlag, ch. Implementation of the Straightening Algorithm, 36–45.

WONG, R., AND HALL, E. 1978. Scene matching with invariant moments. *Computer Graphics and Image Processing 8*, 16–24.

ZAKARIA, M., VROOMEN, L., ZSOMBOR-MURRAY, P., AND VAN KESSEL, J. 1987. Fast algorithm for the computation of moment invariants. *Pattern Recognition 20*, 639–643.