J. Math. Imaging and Vision, 10, 75–85 (1999)

Minimal decomposition of model-based invariants

DAPHNA WEINSHALL AND MICHAEL WERMAN

daphna@cs.huji.ac.il

Institute of Computer Science
The Hebrew University of Jerusalem
91904 Jerusalem, Israel

;

Abstract.
Model-based invariants are relations between model parameters and image measurements, which are independent of the imaging parameters. Such relations are true for all images of the model. Here we describe an algorithm which, given L independent model-based polynomial invariants describing some shape, will provide a linear re-parameterization of the invariants. This re-parameterization has the properties that: (i) it includes the *minimal* number of terms, and (ii) the shape terms are *the same* in all the model-based invariants. This final representation has 2 main applications: (1) it gives new representations of shape in terms of hyperplanes, which are convenient for object recognition; (2) it allows the design of new linear shape from motion algorithms. In addition, we use this representation to identify object classes that have universal invariants.

Keywords: model-based invariants, universal invariants, projective reconstruction

1. Introduction

An image provides us with relations between 3 different kind of parameters: image measurements, shape parameters, and imaging parameters (e.g., camera parameters). Here we restrict ourselves to the domain of multiple points in multiple frames, where the image measurements are $2D$ point coordinates, and the shape parameters are $3D$ point coordinates. There has been much interest in relations involving only image measurements and imaging parameters (e.g., the epipolar geometry and the essential matrix [8]). In this paper we are interested in the dual relations involving only shape measurements and imaging parameters, which are called model-based invariants.

The analysis of invariants arose much interest in computer vision and pattern recognition. As we use the concept here, an invariant is a relation between the image measurements and the model (or shape) parameters. This relation does not depend on variables of the imaging process, such as the camera orientation (viewing position). There are 2 types of invariants:

Model-free invariant: there exist image measurements which always identify the object, so that their value is completely determined by the object regardless of the details of the imaging process. This is usually called an invariant in the literature. Such invariant relations do not exist for $2D$ images of general $3D$ objects [1, 10]; for special classes of objects, such as planar or symmetrical objects, invariant relations may be found [10, 12].

Model-based invariant: a relation which includes mixed terms, representing image measurements or model parameters. Model-based invariant relations exist for many interesting cases [16].

Typically, model-based invariants are complex polynomial relations between the known image measurements and the unknown shape parameters. In the external calibration literature, where the dual relations between camera parameters and image measurements are ordinarily used, it proved very useful to re-parameterize the original relations in a linear form; this turned the computation of camera parameters into a linear problem. Obviously the problem is linear in some new variables, that could be complex functions of the original camera parameters. Examples are the epipolar geometry with the essential matrix [8] or the fundamental bilinear matrix [9], the trilinear tensor [14, 5], and the quadrilinear tensor [15, 18, 4].

In the context of model-based invariants, this lead us to the following questions:

> Given a model-based invariant, how do we find its "optimal" (or most compact) linear re-parameterization? in other words, how do we rewrite it with the minimal number of terms, that truly reflect the number of degrees of freedom of the system? Given L model-based invariants, what is the *simultaneous* minimal linear decomposition of these relations?

By simultaneous we specifically mean that the shape terms are the same in all relations. The reason we seek such simultaneous decomposition of model-based invariants is that *when the shape terms are the same, all relations can be used simultaneously in applications such as direct shape reconstruction.*

Our attempt to answer these questions was motivated by two applications: one is object recognition, the other is $3D$ reconstruction:

- If shape is reconstructed from images for the sole purpose of recognition, the parametric representation of the shape in the model-based invariants captures all the relevant information on the shape. Our algorithm produces a set of L linear relations with a minimal number of terms (these are the equation's unknowns), say n. We would now need at least $\lceil \frac{n}{L} \rceil$ frames to compute shape. This automatically gives us new linear shape reconstruction algorithms, which are likely to be more robust than other linear algorithms. Often the shape (or depth) can be computed directly from the reconstructed models, as will be shown in the examples below.
- Clearly, for the initial indexing step in object recognition, model-based invariants are most suitable (imaging parameters may be computed and used later during the verification step). More specifically, a model-based invariant represents a relation between image measurements and shape parameters that is true in all images of the model. The database which includes these models may be thought of as a big multi-dimensional table, where an object corresponds to the manifold defined by its model-based invariant, and individual images are points on (or pointers into) these manifolds. In such a framework indexing is quick (but the representation is very space inefficient).
  Our algorithm provides an automatic tool to compute low dimensional linear representations of the model-based invariants; when used, these representations simplify the complexity of the recognition process by reducing the indexing complexity. In other words, the representation of objects is made simpler because simple manifolds - hyperplanes - are stored; recognition is made easier because an image provides a pointer - a point which should lie on the hyperplane representing the object depicted in the image.

One other application motivated our work: model-based invariants have a special meaning when the number of linear terms is no larger than 2. In this case it is possible to separate the shape dependency from the dependency on image measurements, and get a *model-free invariant* - a "true" invariant. Such invariant relations do not normally exist [1, 10] unless the class of objects is restricted [10, 12]. Our theory allows us to identify the kind of class constraints that can be used to reduce the number of linear terms, so that a model-based invariant becomes a model-free invariant.

In this paper we assume that a set of L model-based invariants, which describe the images of some set of objects, is initially given. In Section 3 we describe an algorithm which produces a linear re-parameterization of the invariants, with minimal number of terms. In Section 4 we generalize this algorithm to produces a linear re-parameterization of the L invariants simultaneously, such that at the end the shape terms are the same in all the re-parameterized model-based invariants.

Jacobs [3, 6] studied the complexity of invariants and model-based invariants in their raw form. In [6] he showed that for 6 points in a single perspective image, there exists a nonlinear model-based invariant with 5 unknowns. We obtained a linear re-parameterization with 5 terms, thus the linear representation is no more complex than the nonlinear one in this example. In ECCV '96 [19] we reported a follow up of the basic technique described in Section 3: we united two approaches, the elimination discussed in [18] and the linear re-parameterization of one relation described in Section 3, to accomplish an automatic process that optimizes indexing given a general vision problem.

2. Model-based invariants for n projective points in 1 image

To demonstrate the procedures described in Sections 3,4 we will work out 2 examples in detail, where the model is of 6 or 7 points and the projection model is perspective. We start by using homogeneous coordinates to represent the $3D$ coordinates of the points; thus the representation of the i-th point is $P_i = [X_i, Y_i, Z_i, W_i]^T \in \mathcal{P}^3$. Since we are working in $\mathcal{P}^3$, 5 points define a basis; we select the first 5 points to be a certain (robust) projective basis, leading to the following representation of the $3D$ shape of the points:

$$P_1 = \begin{pmatrix}1\\0\\0\\1\end{pmatrix} P_2 = \begin{pmatrix}0\\1\\0\\1\end{pmatrix} P_3 = \begin{pmatrix}0\\0\\1\\1\end{pmatrix} P_4 = \begin{pmatrix}0\\0\\0\\1\end{pmatrix}$$

$$P_5 = \begin{pmatrix}1\\1\\1\\1\end{pmatrix} P_6 = \begin{pmatrix}X_1\\Y_1\\Z_1\\W_1\end{pmatrix} P_7 = \begin{pmatrix}X_2\\Y_2\\Z_2\\W_2\end{pmatrix}$$

Similarly we use homogeneous coordinates to represent the projected $2D$ coordinates of the points; thus the representation of the i-th image point is $p_i = [x_i, y_i, w_i]^T \in \mathcal{P}^2$. Since we are working in $\mathcal{P}^2$, 4 points define a basis; we select the first 4 points to be the projective basis, leading to the following representation of the image of the points:

$$p_1 = \begin{pmatrix}1\\0\\1\end{pmatrix} p_2 = \begin{pmatrix}0\\1\\1\end{pmatrix} p_3 = \begin{pmatrix}0\\0\\1\end{pmatrix} p_4 = \begin{pmatrix}1\\1\\1\end{pmatrix}$$
$$p_5 = \begin{pmatrix}a_0\\b_0\\c_0\end{pmatrix} p_6 = \begin{pmatrix}a_1\\b_1\\c_1\end{pmatrix} p_7 = \begin{pmatrix}a_2\\b_2\\c_2\end{pmatrix}$$

Given any image of the points, we can always compute the $2D$ projective transformation which will transform the points to the representation given above.

In [2] we showed how to compute model-based invariants in such cases. In this section we review these relations for the special cases of 6 projective points and 7 projective points. Note that the model-based invariants listed below have many terms. Clearly these expressions are of little value for linear reconstruction and indexing unless we can re-parameterize them in a "simpler" way.

2.1. Model-based invariants for 6 projective points

The model-based invariants in this case are obtained from the observation that the following matrix has rank 3 [2, 4], and thus its determinant should be 0:

$$\begin{bmatrix} c_0 & 0 & -a_0 & c_0 - a_0 \\ 0 & c_0 & -b_0 & c_0 - b_0 \\ c_1 X_1 & 0 & -a_1 Z_1 & c_1 W_1 - a_1 W_1 \\ 0 & c_1 Y_1 & -b_1 Z_1 & c_1 W_1 - b_1 W_1 \end{bmatrix}$$

This gives the following constraint (see derivation in [2, 17], cf. with [11]):

$$a_0b_1Z_1Y_1 + c_1{Y_1}^2b_0 - c_1{X_1}^2a_0 - a_0a_1{Z_1}^2 + a_0b_1{Z_1}^2 - b_1{Y_1}^2b_0 + c_0a_1{Z_1}^2 - c_0b_1{Z_1}^2 - b_0a_1{Z_1}^2 + b_0b_1{Z_1}^2 + a_1{X_1}^2a_0 - a_1Y_1a_0X_1 + a_1Y_1a_0Z_1 + c_1Y_1b_0Z_1 - Y_1a_1Z_1c_0 + c_0a_1Z_1X_1 - c_0b_1Z_1Y_1 - b_0a_1Z_1X_1 - a_1Y_1b_0Z_1 - c_1X_1Y_1b_0 + X_1b_1Y_1b_0 + c_1X_1Y_1a_0 + X_1a_0b_1Z_1 - c_1X_1a_0Z_1 + c_0X_1b_1Z_1 - X_1b_0b_1Z_1 + c_1X_1a_0W_1 - c_0a_1Z_1W_1 + c_0b_1Z_1W_1 + a_0a_1Z_1W_1 - a_0b_1Z_1W_1 + b_1Y_1b_0W_1 + b_0a_1Z_1W_1 - b_0b_1Z_1W_1 - c_1Y_1b_0W_1 - a_1X_1a_0W_1 - 2X_1b_1Y_1a_0 + 2a_1Y_1b_0X_1 = 0$$

2.2. Model-based invariants for 7 projective points

The model-based invariants in this case are obtained from the observation that the following matrix has rank 3 [2, 4], and thus each of its 15 4×4 minors should vanish:

$$\begin{bmatrix} c_0 & 0 & -a_0 & c_0 - a_0 \\ 0 & c_0 & -b_0 & c_0 - b_0 \\ c_1\,X_1 & 0 & -a_1\,Z_1 & c_1\,W_1 - a_1\,W_1 \\ 0 & c_1\,Y_1 & -b_1\,Z_1 & c_1\,W_1 - b_1\,W_1 \\ c_2\,X_2 & 0 & -a_2\,Z_2 & c_2\,W_2 - a_2\,W_2 \\ 0 & c_2\,Y_2 & -b_2\,Z_2 & c_2\,W_2 - b_2\,W_2 \end{bmatrix}$$

In [2] we show that there are only 4 algebraic independent constraints which involve all 7 projective points, one of them is the following (the other 3 look similar):

$$-b_1Y_1b_0a_2W_2 - c_0b_1Z_1c_2Z_2 - c_0b_1Z_1c_2Y_2 - c_2X_2c_1Y_1b_0 - c_2X_2b_0b_1Z_1 - c_0b_1Z_1a_2W_2 - a_2X_2b_1Y_1b_0 + a_2X_2a_0c_1X_1 + c_2X_2a_0b_1Z_1 - a_2Y_2b_1Z_1b_0 + c_2X_2b_1Y_1b_0 + a_0b_1Z_1a_2W_2 - a_2X_2a_0b_1Z_1 + a_2X_2b_0b_1Z_1 + b_1Y_1a_2Z_2b_0 + c_0c_2X_2b_1Z_1 - c_0a_2X_2b_1Z_1 + a_2X_2c_1Y_1b_0 + a_0b_1Z_1c_2Z_2 + a_0b_1Z_1c_2Y_2 + a_2Y_2b_1Z_1a_0 + a_2b_0b_1Y_1Y_2 + c_0b_1Z_1c_2W_2 + a_0a_1a_2Z_1Z_2 - a_1a_2c_0Z_1Z_2 + a_1a_2b_0Z_1Z_2 - a_0a_1a_2X_1X_2 - a_0b_1Z_1c_2W_2 + c_2X_2c_1Y_1a_0 - c_2X_2a_0c_1Z_1 + a_2X_2a_0a_1W_1 + a_2X_2a_1Y_1a_0 - c_2X_2a_1Y_1a_0 - a_1Y_1a_2Z_2a_0 + c_2Y_2b_0c_1Y_1 + b_0b_1Z_1a_2W_2 - b_0b_1Z_1c_2W_2 - a_2X_2a_0a_1Z_1 + a_1X_1a_0a_2Z_2 + c_2X_2a_0c_1W_1 - c_2X_2a_0a_1W_1 + a_1X_1b_0a_2Z_2 + a_0a_2Z_2c_1Y_1 - a_0a_2Z_2c_1Z_1 + a_0a_2Z_2c_1W_1 - a_0a_2Z_2a_1W_1 - a_1X_1a_2Y_2b_0 + c_0c_1X_1a_2Z_2 - c_0a_1X_1a_2Z_2 - a_2Y_2a_1Z_1b_0 + c_2X_2a_1Y_1b_0 - a_2X_2c_1Y_1a_0 + a_2X_2a_0c_1Z_1 - a_2X_2a_0c_1W_1 - c_1X_1b_0a_2Z_2 + c_1X_1a_2Y_2b_0 - a_2X_2a_1Y_1b_0 - b_0a_2Z_2c_1Z_1 + b_0a_2Z_2c_1W_1 - c_0a_2Z_2c_1Y_1 + c_0a_2Z_2c_1Z_1 - c_0a_2Z_2c_1W_1 - b_0a_2Z_2a_1W_1 + b_0b_1Z_1c_2Z_2 + b_0b_1Z_1c_2Y_2 - Z_1b_1b_0a_2Z_2 - Z_1b_1a_0a_2Z_2 - a_2Y_2b_1Z_1c_0 - b_1Y_1b_0c_2Z_2 - b_1Y_1b_0c_2Y_2 - c_1X_1a_0a_2Z_2 + c_2X_2a_0a_1Z_1 + Z_1c_0b_1a_2Z_2 - a_1Y_1b_0c_2Y_2 - a_1Y_1b_0c_2Z_2 + a_1Y_1b_0c_2W_2 - a_1Y_1b_0a_2W_2 + a_1Y_1a_2Z_2c_0 + a_2Y_2b_0c_1Z_1 + a_2Y_2b_0a_1W_1 - a_2Y_2b_0c_1W_1 + c_0a_2Z_2a_1W_1 + c_1Y_1b_0a_2W_2 - c_1Y_1b_0c_2W_2 + c_1Y_1b_0c_2Z_2 + a_1X_1a_0c_2X_2 - c_1X_1a_0c_2X_2 + b_1Y_1b_0c_2W_2 + 2b_1Y_1a_2Z_2c_0 - 2c_2X_2b_1Y_1a_0 + 2a_2X_2b_1Y_1a_0 - 2b_0a_2Z_2c_1Y_1 + 2a_1Y_1a_2Z_2b_0 - 2b_1Y_1a_2Z_2a_0 = 0$$

3. Minimal linear invariant: one relation

Let $\mathbf{S}$ denote the set of parameters describing the **object shape**. Let $\mathbf{D}$ denote the data - a set of **image measurements**. Let $\mathcal{I} = \{f^l(\mathbf{S},\mathbf{D}) = 0\}_{l=1}^{L}$ denote the set of independent model-based invariants; we assume that each model-based invariant is polynomial.

We start with the simple case where $\mathcal{I}$ includes a single relation $I : f(\mathbf{S},\mathbf{D}) = 0$. We seek a decomposition of $f()$ in the following compact way, explicitly separating image variables $\mathbf{D}$ from shape variables $\mathbf{S}$:

$$f(\mathbf{S},\mathbf{D}) = \sum_{k=1}^{r} g_k(\mathbf{S}) * h_k(\mathbf{D}) = 0 \qquad (1)$$

g_k and h_k are polynomial functions of the shape $\mathbf{S}$ and the image $\mathbf{D}$ respectively. We call (1) the canonical representation of $f(\mathbf{S},\mathbf{D})$. Note that if $f(\mathbf{S},\mathbf{D})$ is algebraic, as we assume here, such a decomposition always exists.

In the simplest case where $r = 2$, (1) would become:

$$f(\mathbf{S},\mathbf{D}) = h_1(\mathbf{D})g_1(\mathbf{S}) + h_2(\mathbf{D})g_2(\mathbf{S})$$

$$\Rightarrow \quad \frac{h_1(\mathbf{D})}{h_2(\mathbf{D})} = -\frac{g_2(\mathbf{S})}{g_1(\mathbf{S})}$$

Thus if $r = 2$, the model-based invariant $f() = 0$ is really a model-free (a "true") invariant.

3.1. Algorithm

The algebraic expression $f(\mathbf{S},\mathbf{D}) = 0$ can always be written as a sum of multiplications since $f()$ is polynomial; we start by arbitrarily choosing one such representation for $f(\mathbf{S},\mathbf{D})$:

$$f(\mathbf{S},\mathbf{D}) \approx \sum_{i=1}^{n}\sum_{j=1}^{m} q_{ij}s_i d_j = \mathbf{s}\cdot Q \cdot \mathbf{d}^T = 0 \quad (2)$$

where q_{ij} are constants, s_i and d_j are distinct products of element of $\mathbf{S}$ and $\mathbf{D}$ respectively, $\mathbf{s} = [s_1,\ldots,s_n]$, $\mathbf{d} = [d_1,\ldots,d_m]$, and Q is the $n\times m$ matrix whose elements are q_{ij}.

Definition 1. Q is the **complexity-matrix** of the relation $f(\mathbf{S},\mathbf{D}) = 0$.

Theorem 1. *The minimal linear decomposition* $f(\mathbf{S},\mathbf{D}) = \sum_{k=1}^{r} g_k(\mathbf{S}) * h_k(\mathbf{D})$ *has* r *terms, where* r *is equal to the rank of the complexity-matrix* Q.

Proof: The theorem follows from (2), the fact that elementary operations on the rows and columns of a matrix are algebraic operations, and because the rank of a matrix is the minimal number of outer products of vectors that sum to the matrix. Note, however, that this representation obtains the minimal number of terms in a limited context, where a term can only be a linear combination of s_i's or d_j's. □

Algorithm to compute the minimal linear model-based invariant:

1. Compute the SVD (or similar) decomposition of $Q = U\Sigma V^T$; the rank of Q is equal to the number of non-0 elements in the diagonal matrix Σ.
2. By construction

$$\begin{aligned} f(\mathbf{S},\mathbf{D}) &= \mathbf{s}U\Sigma V^T\mathbf{d}^T = \mathbf{g}(\mathbf{S})\cdot\mathbf{h}^T(\mathbf{D}) \\ &= \sum_{i=1}^{r} g_i(\mathbf{S})h_i(\mathbf{D}) \end{aligned}$$

where

$$\begin{aligned} \mathbf{g}(\mathbf{S}) &= [g_1(\mathbf{S}),\ldots,g_r(\mathbf{S})] = \mathbf{s}U\sqrt{\Sigma} \\ \mathbf{h}(\mathbf{D}) &= [h_1(\mathbf{D}),\ldots,h_r(\mathbf{D})] = \mathbf{d}V\sqrt{\Sigma} \end{aligned}$$

Although the rank of Q is unique, the decomposition above is not; other expressions can all be derived using the same SVD decomposition. For example, we can decompose the complexity matrix Q as $Q = (U\sqrt{\Sigma}H)(H^{-1}(\sqrt{\Sigma}V)^T)$, where H denotes any regular $r\times r$ matrix. Now $f(\mathbf{S},\mathbf{D}) = \mathbf{g}'(\mathbf{S})\cdot\mathbf{h}'^T(\mathbf{D})$, where $\mathbf{g}'(\mathbf{S}) = \mathbf{s}U\sqrt{\Sigma}H$ and $\mathbf{h}'(\mathbf{D}) = \mathbf{d}V\sqrt{\Sigma}H$. Thus there are essentially $r^2 - 1$ (-1 from homogeneity) independent decompositions of type (1).

3.2. Example: 6 projective points

We use the notations of Section 2 and the definitions above, where:

- $\mathbf{S}$ denotes the set of 4 shape variables X_1, Y_1, Z_1, W_1, - the $3D$ projective coordinates of the 6th point.
- $\mathbf{D}$ denotes the set of 6 image variables a_0, b_0, c_0, a_1, b_1, c_1; these are the image measurements - the projective image coordinates of the points.

In this case we get the following model-based invariant (see Section 2.1):

$f(\mathbf{S},\mathbf{D}) = a_0b_1Z_1Y_1 + c_1{Y_1}^2b_0 - c_1{X_1}^2a_0 - a_0a_1{Z_1}^2 + a_0b_1{Z_1}^2 - b_1{Y_1}^2b_0 + c_0a_1{Z_1}^2 - c_0b_1{Z_1}^2 - b_0a_1{Z_1}^2 + b_0b_1{Z_1}^2 + a_1{X_1}^2a_0 - a_1Y_1a_0X_1 + a_1Y_1a_0Z_1 + c_1Y_1b_0Z_1 - Y_1a_1Z_1c_0 + c_0a_1Z_1X_1 - c_0b_1Z_1Y_1 - b_0a_1Z_1X_1 - a_1Y_1b_0Z_1 - c_1X_1Y_1b_0 + X_1b_1Y_1b_0 + c_1X_1Y_1a_0 + X_1a_0b_1Z_1 - c_1X_1a_0Z_1 + c_0X_1b_1Z_1 - X_1b_0b_1Z_1 + c_1X_1a_0W_1 - c_0a_1Z_1W_1 + c_0b_1Z_1W_1 + a_0a_1Z_1W_1 - a_0b_1Z_1W_1 + b_1Y_1b_0W_1 + b_0a_1Z_1W_1 - b_0b_1Z_1W_1 - c_1Y_1b_0W_1 - a_1X_1a_0W_1 - 2X_1b_1Y_1a_0 + 2a_1Y_1b_0X_1 = 0$

In order to represent $f(\mathbf{S},\mathbf{D})$ as a sum of multiplications as required in (2), we observe the following:

1. There are 10 shape monomials s_i, thus $n = 10$ and

$$\mathbf{s} = [X_1^2, X_1Y_1, X_1Z_1, X_1W_1, Y_1^2, Y_1Z_1, Y_1W_1, Z_1^2, Z_1W_1, W_1^2] \quad (3)$$

2. There are 9 image monomials d_j, thus $m = 9$ and

$$\mathbf{d} = [a_0a_1, a_0b_1, a_0c_1, b_0a_1, b_0b_1, b_0c_1, c_0a_1, c_0b_1, c_0c_1] \quad (4)$$

3. The complexity matrix Q is 10×9, where $Q[i,j]$ is the coefficient of $s_i d_j$ in the expression above. For example, from the first term in the expression above $Q[6,2] = 1$. Thus:

$$Q^T = \begin{bmatrix} 1 & -1 & 0 & -1 & 0 & 1 & 0 & -1 & 1 & 0 \\ 0 & -2 & 1 & 0 & 0 & 1 & 0 & 1 & -1 & 0 \\ -1 & 1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & -1 & 0 & 0 & -1 & 0 & -1 & 1 & 0 \\ 0 & 1 & -1 & 0 & -1 & 0 & 1 & 1 & -1 & 0 \\ 0 & -1 & 0 & 0 & 1 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

We use Gaussian elimination to decompose Q as $Q = UV^T$, where U is 10×5, and V is 9×5. Many matrices U satisfy these conditions, and we choose a relatively "simple" one:

$$\begin{aligned} Q^T &= VU^T \qquad (5) \\ V &= \begin{bmatrix} 1 & 1/2 & 0 & 0 & -1/2 \\ 0 & 1 & -1 & 0 & 1/2 \\ -1 & -1/2 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 & -1/2 \\ 0 & -1/2 & 1 & 1 & 1/2 \\ 0 & 1/2 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & 1/2 \\ 0 & 0 & -1 & 0 & -1/2 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\ U^T &= \begin{bmatrix} 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -2 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 & 0 \end{bmatrix} \end{aligned}$$

Since the rank of Q is 5, we can rewrite the model-based invariant $f(\mathbf{S},\mathbf{D})$ as follows:

$$f(\mathbf{S},\mathbf{D}) = \mathbf{g}(\mathbf{S}) \cdot \mathbf{h}^T(\mathbf{D}) = \sum_{i=1}^{5} g_i(\mathbf{S}) h_i(\mathbf{D}) = 0 \quad (6)$$

where (3), (5) give

$$\mathbf{g}(\mathbf{S}) = \mathbf{s} \cdot U = \begin{pmatrix} X_1^2 - X_1 W_1 \\ -2X_1Y_1 + 2Y_1Z_1 \\ -X_1Z_1 + Y_1Z_1 \\ -Y_1^2 + Y_1W_1 \\ 2Z_1^2 - 2Z_1W_1 \end{pmatrix} \quad (7)$$

and (4), (5) give

$\mathbf{h}(\mathbf{D}) = \mathbf{d} \cdot V = [a_1a_0 - c_1a_0\ ,\ 0.5a_1a_0 + a_0b_1 - 0.5c_1a_0 - b_0a_1 - 0.5b_0b_1 + 0.5b_0c_1\ ,\ -a_0b_1 + c_1a_0 + b_0a_1 + b_0b_1 - c_0a_1 - c_0b_1\ ,\ b_0b_1 - b_0c_1\ ,\ -0.5a_1a_0 + 0.5a_0b_1 - 0.5b_0a_1 + 0.5b_0b_1 + 0.5c_0a_1 - 0.5c_0b_1]$

4. Minimal linear invariant: multiple relations

Let $\mathbf{S}$, $\mathbf{D}$ and $\mathcal{I} = \{f^l(\mathbf{S},\mathbf{D}) = 0\}_{l=1}^{L}$ as in Section 3. We now consider the general case where $\mathcal{I}$ includes $L > 1$ relations. We look for a simultaneous decomposition of the L relations, such that they have a minimal number of terms, and the shape terms are identical in all the relations. More specifically, we look for a simultaneous decomposition:

$$f^l(\mathbf{S},\mathbf{D}) = \sum_{k=1}^{r} g_k(\mathbf{S}) * h_k^l(\mathbf{D}) = 0$$

where g_k and h_k^l are polynomial functions of the shape $\mathbf{S}$ and the image $\mathbf{D}$ respectively. Note that $g_k(\mathbf{S})$, the shape terms, do not depend on the index l - this is what we mean by simultaneous decomposition of the L relations.

4.1. Algorithm

We start by writing each algebraic expression $f^l(\mathbf{S},\mathbf{D}) = 0$ as a sum of multiplications:

$$f^l(\mathbf{S},\mathbf{D}) \approx \sum_{i=1}^{n}\sum_{j=1}^{m} q_{ij}^l s_i d_j = \mathbf{s} \cdot Q^l \cdot \mathbf{d}^T = 0 \quad (8)$$

where s_i and d_j are distinct products of element of $\mathbf{S}$ and $\mathbf{D}$ respectively, $\mathbf{s} = [s_1, \ldots, s_n]$, $\mathbf{d} = [d_1, \ldots, d_m]$, and Q^l is the $n \times m$ matrix whose elements are q_{ij}^l. From Def. 1, Q^l is the **complexity-matrix** of the relation $f^l(\mathbf{S},\mathbf{D}) = 0$.

Definition 2. Q, the matrix obtained by concatenating the L matrices Q^l from left to right, is the **joint complexity-matrix** of $\mathcal{I}$.

Note that the size of Q is $n \times Lm$. Note also the asymmetrical role of rows and columns here: the row variables are shape variables, and thus should be the same for all invariants; the column variables are data variables, and thus can (and should) vary for different invariants.

Theorem 2. *The minimal simultaneous linear decomposition $f^l(\mathbf{S}, \mathbf{D}) = \sum_{k=1}^{r} g_k(\mathbf{S}) * h_k(\mathbf{D})$ has r terms, where r is equal to the rank of the joint complexity-matrix Q.*

The proof is similar to the proof of theorem 1, with the same restriction: the number of terms is minimal in a limited context, where a term can only be a linear combination of s_i's or d_j's.

We can now derive the following *algorithm* to compute the minimal simultaneous linear decomposition of model-based invariants:

1. Compute the SVD (or similar) decomposition of the joint complexity matrix $Q = U\Sigma V^T$.
2. For each invariant, find a decomposition $Q^l = U(V^l)^T$; below we specifically use $(V^l)^T = (U\Sigma)^+ Q^l$, where $(U\Sigma)^+$ denotes the pseudo-inverse of $(U\Sigma)$.
3. By construction

$$\begin{aligned} f^l(\mathbf{S}, \mathbf{D}) &= \mathbf{s}U(V^l)^T\mathbf{d}^T = \mathbf{g}(\mathbf{S}) \cdot (\mathbf{h}^l(\mathbf{D}))^T \\ &= \sum_{i=1}^{r} g_i(\mathbf{S}) h_i^l(\mathbf{D}) \end{aligned}$$

where

$$\begin{aligned} \mathbf{g}(\mathbf{S}) &= [g_1(\mathbf{S}), \ldots, g_r(\mathbf{S})] = \mathbf{s}U \\ \mathbf{h}^l(\mathbf{D}) &= [h_1^l(\mathbf{D}), \ldots, h_r^l(\mathbf{D})] = \mathbf{d}V^l \\ &= \mathbf{d}(Q^l)^T U\Sigma((U\Sigma)^T U\Sigma)^{-1} \end{aligned}$$

4.2. Example: 7 projective points

We use the notations of Section 2 and the definitions of Section 3, where:

- $\mathbf{S}$ denotes the set of 8 shape variables X_1, Y_1, Z_1, W_1, X_2, Y_2, Z_2, W_2 - the $3D$ projective coordinates of the 6th and 7th points.
- $\mathbf{D}$ denotes the set of 9 image variables a_0, b_0, c_0, a_1, b_1, c_1, a_2, b_2, c_2; these are the image measurements - the projective image coordinates of the points.

In this case we have 4 independent model-based invariants (see derivation in Section 2.2); one is given below, the other 3 look similar and are therefore omitted.

$$\begin{aligned} f^1(\mathbf{S}, \mathbf{D}) = {} & -b_1Y_1b_0a_2W_2 - c_0b_1Z_1c_2Z_2 - c_0b_1Z_1c_2Y_2 - \\ & c_2X_2c_1Y_1b_0 - c_2X_2b_0b_1Z_1 - c_0b_1Z_1a_2W_2 - a_2X_2b_1Y_1b_0 + \\ & a_2X_2a_0c_1X_1 + c_2X_2a_0b_1Z_1 - a_2Y_2b_1Z_1b_0 + c_2X_2b_1Y_1b_0 + \\ & a_0b_1Z_1a_2W_2 - a_2X_2a_0b_1Z_1 + a_2X_2b_0b_1Z_1 + b_1Y_1a_2Z_2b_0 + \\ & c_0c_2X_2b_1Z_1 - c_0a_2X_2b_1Z_1 + a_2X_2c_1Y_1b_0 + a_0b_1Z_1c_2Z_2 + \\ & a_0b_1Z_1c_2Y_2 + a_2Y_2b_1Z_1a_0 + a_2b_0b_1Y_1Y_2 + c_0b_1Z_1c_2W_2 + \\ & a_0a_1a_2Z_1Z_2 - a_1a_2c_0Z_1Z_2 + a_1a_2b_0Z_1Z_2 - a_0a_1a_2X_1X_2 - \\ & a_0b_1Z_1c_2W_2 + c_2X_2c_1Y_1a_0 - c_2X_2a_0c_1Z_1 + a_2X_2a_0a_1W_1 + \\ & a_2X_2a_1Y_1a_0 - c_2X_2a_1Y_1a_0 - a_1Y_1a_2Z_2a_0 + c_2Y_2b_0c_1Y_1 + \\ & b_0b_1Z_1a_2W_2 - b_0b_1Z_1c_2W_2 - a_2X_2a_0a_1Z_1 + a_1X_1a_0a_2Z_2 + \\ & c_2X_2a_0c_1W_1 - c_2X_2a_0a_1W_1 + a_1X_1b_0a_2Z_2 + a_0a_2Z_2c_1Y_1 - \\ & a_0a_2Z_2c_1Z_1 + a_0a_2Z_2c_1W_1 - a_0a_2Z_2a_1W_1 - a_1X_1a_2Y_2b_0 + \\ & c_0c_1X_1a_2Z_2 - c_0a_1X_1a_2Z_2 - a_2Y_2a_1Z_1b_0 + c_2X_2a_1Y_1b_0 - \\ & a_2X_2c_1Y_1a_0 + a_2X_2a_0c_1Z_1 - a_2X_2a_0c_1W_1 - c_1X_1b_0a_2Z_2 + \\ & c_1X_1a_2Y_2b_0 - a_2X_2a_1Y_1b_0 - b_0a_2Z_2c_1Z_1 + b_0a_2Z_2c_1W_1 - \\ & c_0a_2Z_2c_1Y_1 + c_0a_2Z_2c_1Z_1 - c_0a_2Z_2c_1W_1 - b_0a_2Z_2a_1W_1 + \\ & b_0b_1Z_1c_2Z_2 + b_0b_1Z_1c_2Y_2 - Z_1b_1b_0a_2Z_2 - Z_1b_1a_0a_2Z_2 - \\ & a_2Y_2b_1Z_1c_0 - b_1Y_1b_0c_2Z_2 - b_1Y_1b_0c_2Y_2 - c_1X_1a_0a_2Z_2 + \\ & c_2X_2a_0a_1Z_1 + Z_1c_0b_1a_2Z_2 - a_1Y_1b_0c_2Y_2 - a_1Y_1b_0c_2Z_2 + \\ & a_1Y_1b_0c_2W_2 - a_1Y_1b_0a_2W_2 + a_1Y_1a_2Z_2c_0 + a_2Y_2b_0c_1Z_1 + \\ & a_2Y_2b_0a_1W_1 - a_2Y_2b_0c_1W_1 + c_0a_2Z_2a_1W_1 + c_1Y_1b_0a_2W_2 - \\ & c_1Y_1b_0c_2W_2 + c_1Y_1b_0c_2Z_2 + a_1X_1a_0c_2X_2 - c_1X_1a_0c_2X_2 + \\ & b_1Y_1b_0c_2W_2 + 2b_1Y_1a_2Z_2c_0 - 2c_2X_2b_1Y_1a_0 + 2a_2X_2b_1Y_1a_0 - \\ & 2b_0a_2Z_2c_1Y_1 + 2a_1Y_1a_2Z_2b_0 - 2b_1Y_1a_2Z_2a_0 = 0 \end{aligned}$$

In order to represent $f^1(\mathbf{S}, \mathbf{D})$ as a sum of multiplications as in (2), we observe the following:

1. There are 16 shape monomials s_i, thus $n = 16$ and

$$\mathbf{s} = [X_1X_2, X_1Y_2, X_1Z_2, X_1W_2, Y_1X_2, Y_1Y_2, Y_1Z_2, Y_1W_2, Z_1X_2, \ldots, W_1W_2] \quad (9)$$

2. There are 27 image monomials d_j, thus $m = 27$ and

$$\mathbf{d} = [a_0a_1a_2, a_0a_1b_2, a_0a_1c_2, a_0b_1a_2, \ldots, c_0c_1c_2] \quad (10)$$

3. The complexity matrix Q^1 is 16×27, where $Q^1[i,j]$ is the coefficient of $s_i d_j$ in the expression above. $Q^1[1,1]$, for example, contains the coefficient of $a_0 a_1 a_2 X_1 X_2$ from the expression above, which happens to be -1. Thus we get the following full description of Q^1:

$$
\left[\begin{array}{cccccccccccccc}
-1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & -1 & 2 & 0 & -2 & -1 & 0 & 1 & -1 & 0 & 1 & -1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 \\
-1 & 0 & 0 & -2 & 0 & 0 & 1 & 0 & 0 & 2 & 0 & -1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & -1 & 0 \\
-1 & 0 & 1 & -1 & 0 & 1 & 1 & 0 & -1 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 0 \\
1 & 0 & 0 & -1 & 0 & 1 & -1 & 0 & 0 & 1 & 0 & 0 & -1 & 0 \\
0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
1 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
-1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}\right.
$$

$$
\left.\begin{array}{ccccccccccccc}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & -2 & 0 & 1 & 1 & 0 & 0 & 2 & 0 & 0 & -1 & 0 & 0 \\
1 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 0 & 0 & 0 \\
1 & -1 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & -1 & 1 & 0 & 0 \\
-1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}\right]
$$

Similarly, we rewrite $f^2(\mathbf{S},\mathbf{D})$, $f^3(\mathbf{S},\mathbf{D})$, and $f^4(\mathbf{S},\mathbf{D})$, to construct Q^2, Q^3 and Q^4 in a similar way. The joint complexity matrix Q is constructed by concatenating Q^1, Q^2, Q^3, Q^4, and its size is 16×108.

The rank of Q is computed to be 11. Using Gaussian elimination, we compute a decomposition $Q = UV^T$, where U is 16×11, and V is 108×11. Many matrices U satisfy these conditions, and we choose the following relatively "simple" one:

$$
U = \begin{bmatrix}
-1 & 0 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -1 & 0 & 2 & 0 & 2 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 2 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -2 & 0 & -1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix} \quad (11)
$$

U is used to obtain the individual decompositions:

$$Q^i = U(V^i)^T \implies (V^i)^T = (U^T U)^{-1} U^T Q^i \quad (12)$$

Finally, we rewrite the model-based invariants $f^l(\mathbf{S},\mathbf{D})$, $l = 1..4$. For example, take $f^1(\mathbf{S},\mathbf{D})$:

$$f^1(\mathbf{S},\mathbf{D}) = \mathbf{g}(\mathbf{S}) \cdot (\mathbf{h}^1(\mathbf{D}))^T = \sum_{i=1}^{11} g_i(\mathbf{S}) h_i^1(\mathbf{D}) = 0$$

where (9) and (11) give

$$
\mathbf{g}(\mathbf{S}) = [s_1, \ldots, s_{16}] \cdot U = \begin{pmatrix}
-X_1X_2 + W_1X_2 \\
-X_1Y_2 + Y_1X_2 \\
X_1Z_2 - Y_1Z_2 \\
-2X_1X_2 + 2X_1W_2 \\
-2Y_1X_2 + 2Y_1Z_2 \\
-Y_1Y_2 + Y_1W_2 \\
2Y_1Z_2 - 2Z_1Y_2 \\
2Y_1W_2 - 2W_1Y_2 \\
Z_1X_2 - Z_1Y_2 \\
-2Z_1Z_2 + 2Z_1W_2 \\
-2Z_1W_2 + 2W_1Z_2
\end{pmatrix}
$$

and (10), (11), (12) give

$\mathbf{h}^1(\mathbf{D}) = [d_1, \ldots, d_{27}] \cdot V^1 = [a_0a_1a_2 - a_1a_0c_2 - a_2a_0c_1 + c_1a_0c_2\ ,\ b_0a_1a_2 - b_0c_1a_2\ ,\ a_0a_1a_2 - a_2a_0c_1 + b_0a_1a_2 - b_0c_1a_2 - c_0a_1a_2 + c_0c_1a_2\ ,\ 0\ ,\ -\frac{1}{2}a_0a_1a_2 + \frac{1}{2}a_1a_0c_2 - a_0b_1a_2 + a_0b_1c_2 + \frac{1}{2}a_2a_0c_1 - \frac{1}{2}c_1a_0c_2 + b_0a_1a_2 - \frac{1}{2}b_0a_1c_2 + \frac{1}{2}b_0b_1a_2 - \frac{1}{2}b_0b_1c_2 - b_0c_1a_2 + \frac{1}{2}b_0c_1c_2\ ,\ b_0a_1c_2 - b_0b_1a_2 + b_0b_1c_2 -$

$b_0c_1c_2$, $\frac{1}{2}a_0a_1a_2 - \frac{1}{2}a_1a_0c_2 - a_0b_1c_2 - \frac{1}{2}a_2a_0c_1 + \frac{1}{2}c_1a_0c_2 + \frac{1}{2}b_0a_1a_2 - \frac{1}{2}b_0c_1a_2 + c_0b_1a_2$, $-\frac{1}{2}b_0a_1a_2 + \frac{1}{2}b_0c_1a_2$, $-a_0a_1a_2 + a_1a_0c_2 - a_0b_1a_2 + a_0b_1c_2 + a_2a_0c_1 - c_1a_0c_2 + b_0b_1a_2 - b_0b_1c_2 - c_0b_1a_2 + c_0b_1c_2$, $-\frac{1}{2}a_0a_1a_2 + \frac{1}{2}a_0b_1a_2 - \frac{1}{2}a_0b_1c_2 + \frac{1}{2}a_2a_0c_1 - \frac{1}{2}b_0a_1a_2 + \frac{1}{2}b_0b_1a_2 - \frac{1}{2}b_0b_1c_2 + \frac{1}{2}b_0c_1a_2 + \frac{1}{2}c_0a_1a_2 - \frac{1}{2}c_0b_1a_2 + \frac{1}{2}c_0b_1c_2 - \frac{1}{2}c_0c_1a_2$, $-\frac{1}{2}a_0a_1a_2 + \frac{1}{2}a_2a_0c_1 - \frac{1}{2}b_0a_1a_2 + \frac{1}{2}b_0c_1a_2 + \frac{1}{2}c_0a_1a_2 - \frac{1}{2}c_0c_1a_2$]

5. Adding class constraints

Once it has been shown that model-free invariants do not exist for unconstrained objects [1, 10], attention had turned to characterizing the constraints (or classes of objects) which would lead to model-free invariants [10, 12]. The present analysis allows us to ask this question as part of a more general problem: what class constraints on objects reduce the number of terms in the minimal linear decomposition? In this section we determine sufficient conditions on class constraints to reduce the number of terms, in particular to reduce it to 2 (implying the existence of model-free invariants).

We start from a relation

$$f(\mathbf{S},\mathbf{D}) = \sum_{k=1}^{r} g_k(\mathbf{S}) * h_k(\mathbf{D}) = 0$$

where $g_k(\mathbf{S})$ and $h_k(\mathbf{D})$ are polynomial functions of the shape and image measurements respectively. Every class constraint of the form $\lambda(\mathbf{S}) = 0$, where $\lambda(\mathbf{S})$ divides some $\sum \alpha_k g_k(\mathbf{S})$, reduces the number of terms in the minimal decomposition by at least 1. Thus:

Theorem 3. (class constraints:) *To reduce the minimal number of terms from r to $p < r$, the class constraints should provide at most $(r-p)$ independent constraints of the form $\lambda_i(\mathbf{S}) = 0$, where each $\lambda_i(\mathbf{S})$ divides some $\sum \alpha_k g_k$ modulo the $\lambda_j(\mathbf{S})$, $j < i$.*

Clearly there is a tradeoff between complexity (the number of terms), which is higher for more general (and less constrained) classes, and the density of the database, which is smaller for more general classes (as there are fewer types of such general objects).

Example: given 6 points and a perspective camera, $r = 5$ (see Section 3.2).

- From the minimal model-based invariant developed in Section 3.2, and the theorem above, it immediately follows that if any of the parameters of the 6th point, X_1, Y_1, Z_1, W_1, equals 0, then $r = 3$; if any 2 parameters of the 6th point are equal, then $r = 3$. Thus *if 4 of the 6 points are coplanar, the number of terms in the minimal model-based invariant is 3.*
- If 2 pairs of the 4 parameters X_1, Y_1, Z_1, W_1 are equal then $r = 2$, namely, there is a model-free invariant. The geometry of this case is as follows: one point lies on the line of intersection of 2 of the planes, each spanned by triplets of the remaining 5 points.

6. Reconstruction example: lab sequence

We use a real sequence of images from the 1991 motion workshop, which includes 16 images of a robotic laboratory obtained by rotating a robot arm 120^o (one frame is shown in Fig. 1). 32 corner-like points were tracked. The depth values of the points in the first frame ranged from 13 to 33 feet; moreover, a wide-lens camera was used, causing distortions at the periphery which were not compensated for. (See a more detailed description in [13] Fig. 4, or [7] Fig. 3.)

We compute the shape of the tracked points as follows. We first choose an arbitrary basis of 5 points; for each additional point we:

1. compute $\mathbf{g}(\mathbf{S})$ as define in (7), using all the available frames to solve an over-determined linear system of equations, where each frame provides the constraint given in (6).
2. compute the homogeneous coordinates of the 6th point $[\tilde{X}, \tilde{Y}, 1, \tilde{W}]$ from $\mathbf{g}(\mathbf{S})$ using

$$\begin{aligned} \tilde{X} &= \frac{2\mathbf{g}_1(\mathbf{S}) + \mathbf{g}_2(\mathbf{S}) - 2\mathbf{g}_3(\mathbf{S})}{\mathbf{g}_5(\mathbf{S}) - 2\mathbf{g}_3(\mathbf{S})} \\ \tilde{Y} &= \frac{\mathbf{g}_2(\mathbf{S}) - 2\mathbf{g}_4(\mathbf{S})}{\mathbf{g}_5(\mathbf{S}) + 2\mathbf{g}_3(\mathbf{S})} \\ \tilde{W} &= 1 - \frac{(\mathbf{g}_2(\mathbf{S}) - 2\,\mathbf{g}_4(\mathbf{S}))\,\mathbf{g}_5(\mathbf{S})}{(2\,\mathbf{g}_5(\mathbf{S}) + 4\,\mathbf{g}_3(\mathbf{S}))\,\mathbf{g}_3(\mathbf{S})} + \\ &\quad \frac{(2\,\mathbf{g}_1(\mathbf{S}) + \mathbf{g}_2(\mathbf{S}) - 2\,\mathbf{g}_3(\mathbf{S}))\,\mathbf{g}_5(\mathbf{S})}{(2\,\mathbf{g}_5(\mathbf{S}) - 4\,\mathbf{g}_3(\mathbf{S}))\,\mathbf{g}_3(\mathbf{S})} \end{aligned}$$

3. in order to compare the results with the real $3D$ shape of the points, we multiply the projective homogeneous coordinates by the actual

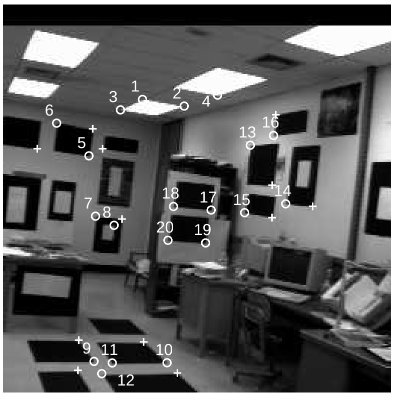

Fig. 1. One frame from the lab sequence.

$3D$ coordinates of the projective basis points, to obtain the equivalent Euclidean representation.

The real $3D$ coordinates of about half the points in the sequence, and the corresponding reconstructed $3D$ coordinates, are the following:

real shape:

$$\begin{bmatrix} -0.3 & -1.7 & -0.3 & 1.8 & 5.3 & 9.9 & 3.2 & -2.3 & 1.5 & -0.6 & 0.5 & 1.5 & -0.5 \\ -4 & -2.6 & 4.4 & 6.3 & 4.2 & -1.6 & -2.8 & -2 & 5 & 3 & 2 & 0.9 & 1 \\ 16.4 & 17.1 & 19.7 & 20 & 25.3 & 29.8 & 31.6 & 15.1 & 21.7 & 21.5 & 21.6 & 21 & 21.6 \end{bmatrix}$$

reconstructed shape:

$$\begin{bmatrix} -0.3 & -1.8 & -0.6 & 0.9 & 3.8 & 8.9 & -0.8 & -2.4 & 0.7 & -0.6 & 0.3 & 1.3 & -0.4 \\ -3.6 & -1.3 & 5 & 6.2 & 4.4 & -1.5 & 0.7 & -1.3 & 4.8 & 2.6 & 1.8 & 0.4 & 0.7 \\ 15.8 & 14.7 & 21.5 & 24.9 & 27.5 & 30.1 & 9.7 & 13.4 & 23.5 & 20.2 & 21.1 & 21.1 & 19.2 \end{bmatrix}$$

The median relative error, where the relative error is the error at each point divided by the distance of the point from the origin, is 12%.

7. Summary

We described an automatic process to simplify model-based invariants by re-parameterizing them in a linear way, and with a minimal number of terms. We demonstrated this process on 2 examples, using model-based invariants of 6 and 7 points under perspective projection. Thus, for example, we obtained 4 homogeneous linear equations with 11 unknowns using the invariants of 7 points. We can use these invariants to compute the shape of the 7 points with a linear algorithm, using at least 3 frames and least squares optimization (since the data is redundant).

Acknowledgements

This research was supported by the Israeli Ministry of Science under Grant 032.7568. Vision research at the Hebrew University is supported by the U.S. Office of Naval Research under Grant N00014-93-1-1202, R&T Project Code 4424341—01.

References

1. J.B. Burns, R. Weiss, and E. Riseman. View variation of point-set and line segment features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(1):51–68, 1993.
2. S. Carlsson and D. Weinshall. Duality of reconstruction and positioning from projective views. *International Journal of Computer Vision*, 1997. in press.
3. D. T. Clemens and D. W. Jacobs. Space and time bounds on indexing 3-D models from 2-D images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(10):1007–1017, 1991.
4. O. Faugeras and B. Mourrain. On the geometry and algebra of the point and line correspondences between N images. In *Proceeding of the Europe-China Workshop on geometrical modeling and invariant for computer vision.* Xidian University Press, 1995.
5. R. Hartley. Lines and points in three views - an integrated approach. In *Proceedings Image Understanding Workshop*, pages 1009–10016, San Mateo, CA, 1994. Morgan Kaufmann Publishers, Inc.
6. D. W. Jacobs. Matching 3-D models to 2-D images. *International Journal of Computer Vision*, 21(1):123-153, January 1997.
7. R. Kumar and A. R. Hanson. Sensitivity of the pose refinement problem to accurate estimation of camera parameters. In *Proceedings of the 3rd International Conference on Computer Vision*, pages 365–369, Osaka, Japan, 1990. IEEE, Washington, DC.

8. H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, 1981.
9. Q.-T. Luong and O. Faugeras. The fundamental matrix: theory, algorithms, and stability analysis. *International Journal of Computer Vision*, 17(1): 43-75, January 1996.
10. Y. Moses and S. Ullman. Limitations of non model-based recognition schemes. In G. Sandini, editor, *Proc. 2nd European Conf. on Computer Vision, Lecture Notes in Computer Science*, volume 588, pages 820–828. Springer Verlag, 1992.
11. L. Quan. Invariants of 6 points from 3 uncalibrated images. In *Proceedings of the 3rd European Conference on Computer Vision*, pages 459–470, Stockholdm, Sweden, 1994. Springer-Verlag.
12. C.A. Rothwell, D.A. Forsyth, A. Zisserman, and J.L. Mundy. Extracting projective structure from single perspective views of 3d point sets. In *Proceedings of the 4th International Conference on Computer Vision*, pages 573–582, Berlin, Germany, 1993. IEEE, Washington, DC.
13. H. S. Sawhney, J. Oliensis, and A. R. Hanson. Description and reconstruction from image trajectories of rotational motion. In *Proceedings of the 3rd International Conference on Computer Vision*, pages 494–498, Osaka, Japan, 1990. IEEE, Washington, DC.
14. A. Shashua. Trilinearity in visual recognition by alignment. In *Proceedings of the 3rd European Conference on Computer Vision*, Stockholm, Sweden, May 1994.
15. B. Triggs. The geometry of projective reconstruction I: matching constraints and the joint image. Technical report, LIFIA, INRIA Rhone Alpes, 1994.
16. D. Weinshall. Model-based invariants for 3D vision. *International Journal of Computer Vision*, 10(1):27–42, 1993.
17. D. Weinshall, M. Werman, and A. Shashua. Shape tensors for efficient and learnable indexing. In *Proceedings of the IEEE Workshop on Representations of Visual Scenes*, Cambridge, Mass, 1995.
18. M. Werman and A. Shashua. Elimination: An approach to the study of 3D-from-2D. In *Proceedings of the 1st International Conference on Computer Vision*, June 1995.
19. M. Werman and D. Weinshall. Complexity of indexing: Efficient and learnable large database indexing. In *Proceedings of the 4th European Conference on Computer Vision*, Cambridge, UK, 1996. Springer-Verlag.

Daphna Weinshall received the BSc degree in Mathematics and computer science from Tel-Aviv University, Tel-Aviv, Israel, in 1982. She received the MSc and PhD degrees in Mathematics and statistics from Tel-Aviv University in 1985 and 1986, working on models of evolution and population genetics. Between 1987 and 1992 she visited the Computational Neuroscience Laboratories at New-York University, did postdoctoral research at the center for biological information processing, MIT, and visited IBM T.J. Watson Research Center. In 1993 she joined the Institute of Computer Science at the Hebrew University of Jerusalem, where she is now a senior lecturer. She was awarded the Weizmann postdoctoral fellowship in 1987, the Alon fellowship in 1993, and an Industrial young investigators research award in 1995. Dr. Weinshall's research interests include computer and biological vision, machine and perceptual and learning. She published papers in qualitative vision, visual psychophysics, neural networks, Bayesian vision, invariants, multi-point and multi-frame geometry, and image and model point-based metrics. Selected publications can be obtained from http://www.cs.huji.ac.il/ daphna.