

Efficient Invariant Representations

PETER MEER

Department of Electrical and Computer Engineering, Rutgers University, P.O. Box 909, Piscataway, NJ 08855-0909, USA

REINER LENZ

Image Processing Laboratory, Department of Electrical Engineering, Linköping University, S-58183 Linköping, Sweden

SUDHIR RAMAKRISHNA

Department of Electrical and Computer Engineering, Rutgers University, P.O. Box 909, Piscataway, NJ 08855-0909, USA

Received September 4, 1995; Revised March 12, 1996; Accepted July 4, 1996

Abstract. Invariant representations are frequently used in computer vision algorithms to eliminate the effect of an unknown transformation of the data. These representations, however, depend on the order in which the features are considered in the computations. We introduce the class of projective/permutation p^2 -invariants which are insensitive to the labeling of the feature set. A general method to compute the p^2 -invariant of a point set (or of its dual) in the *n*-dimensional projective space is given. The one-to-one mapping between n + 3 points and the components of their p^2 -invariant representation makes it possible to design correspondence algorithms with superior tolerance to positional errors. An algorithm for coplanar points in projective correspondence is described as an application, and its performance is investigated. The use of p^2 -invariants as an indexing tool in object recognition systems may also be of interest.

Keywords: invariance, feature correspondence, indexing, geometric hashing

1. Introduction

The following generic problem often appears during the execution of many computer vision tasks. Two sets of features are given, called in the sequel the *reference* and the *transformed set*. An unknown transformation maps a subset of reference features into a subset of the transformed set. It is required to match the corresponding features between the two sets. In the *correspondence problem*, central to motion and stereo tasks, this issue is directly addressed. However the *indexing problem*, i.e., identifying from a database the model best representing the data, can also be regarded as belonging to the same class. For real images the explicit correspondence problem usually is solved making use of metrical properties, i.e., the correspondence is obtained from proximity relations. Salient features (most often corners) are detected in both images. The two feature sets are related to identical coordinate systems and the correspondences are sought in windows defined around (say) the reference feature locations. Whenever the two images are taken from very different viewpoints (the two feature sets are connected by a strong affine or projective transformation) the search window based correspondence methods will yield many false matches. In this case the windows have to be large to account for the changes between the two images and numerous candidates are included. For an extensive application of this technique see for example (Zhang et al., 1994).

The role of metric information is different in the solutions proposed for the indexing problem. For example, in the *alignment* type methods (Huttenlocher and Ullman, 1990), the most proper transformation between the two sets is sought by randomly sampling the space of possible transformations. For a given transformation, the transformed set is backprojected onto the reference set and the local deviations are used to build a performance measure. Alignment implies a correct transformation, and thus a correct match between the points is used to define that transformation. The random sampling of the transform space is eliminated by using invariant representations for the two sets. Geometric hashing (Lamdan et al., 1990) is an example of the invariance based approach toward the indexing problem.

The method described in this paper belongs to the class of invariant representations, and treats the correspondence and indexing problems as equivalent. Its application to object recognition is discussed at the end of the paper. Today, invariance is a major research area of computer vision. See the books (Mundy and Zisserman, 1992; Mundy et al., 1994a) for comprehensive surveys of the results. Invariant representations, however, are permutation-sensitive. That is, the value of an invariant associated with a subset of features often depends on the order in which the features were considered in its computation. Permutation sensitivity introduces a combinatorial explosion in the indexing problem. In the *n*-dimensional projective space \mathbf{P}^n an invariant representation is based on (n+3)-tuples of features, and for each (n + 3)-tuple (n + 3)! different labelings have to be considered.

In this paper we describe a general method to obtain representations of point-sets in projective spaces of arbitrary dimension which are insensitive to both projective transformations and permutations of the labeling of the set. They will be called *projective and permutation*, p^2 -invariants. Interchanging the indices of n + 3 features is equivalent to the action of the permutation group on the (n + 3)-tuple. The permutation group has its own permutation invariants, expressions whose values are unchanged by the reordering of the elements. A p^2 -invariant, $I[\cdot]$, thus must satisfy the condition

$$I(\mathbf{r}_1, \dots, \mathbf{r}_k) = I(\pi\{\mathbf{r}_1, \dots, \mathbf{r}_k\})$$
$$= I(\pi\{T(\mathbf{r}_1), \dots, T(\mathbf{r}_k)\})$$
(1)

where r_k and $T(r_k)$ are the reference and the transform features respectively, and π { $r_1, ..., r_k$ } describes a permutation of the labels.

Invariant representations are very sensitive to noise, i.e., to positional errors in the location of the features. Errors of one or two pixels can drastically change the value of the invariant associated with a subset of features. To reduce the amount of false matches, topological relations among the features (e.g., they are known to belong to the same object located in the front of the camera) are extensively employed. Examples of such object recognition systems are described in (Mundy et al., 1994b; Rothwell, 1994; Zisserman et al., 1995).

The p^2 -invariants are useful beyond removing the dependence on the labeling. In \mathbf{P}^n an (n + 3)-tuple of points can be represented with only *n* numbers, the number of independent projective invariants. However, it will be shown that using a redundant p^2 -representation (based on at least n + 3 numbers), significantly increases the tolerance to positional errors and allows the design of less sensitive correspondence algorithms.

In Section 2 the basic p^2 -invariant of four collinear points is derived. The solution for arbitrary dimensional projective spaces is presented in Section 3. In Section 4 an application in the projective plane \mathbf{P}^2 , a robust method for establishing point correspondences with reduced sensitivity to outliers and positional errors, is described. The case of \mathbf{P}^3 is discussed in Section 5. The consensus principle, repeatedly exploited to increase the tolerance to noise, as well as the use of p^2 invariants in object recognition tasks, are discussed in Section 6.

2. p^2 -Invariant of Four Collinear Points

The fundamental projective invariant is the cross ratio of four collinear points A_i , i = 1, ..., 4. Let the points have homogeneous coordinates $\mathbf{x}_i = (x_1^{(i)}, x_2^{(i)})^t$, then their cross ratio can be defined as

$$I_{1}(\mathbf{x}_{1}, \mathbf{x}_{2}, \mathbf{x}_{3}, \mathbf{x}_{4}) = \lambda = \frac{(A_{1}A_{3})(A_{2}A_{4})}{(A_{3}A_{2})(A_{4}A_{1})}$$
$$= \frac{\begin{vmatrix} x_{1}^{(1)} & x_{1}^{(3)} \\ x_{2}^{(1)} & x_{2}^{(3)} \end{vmatrix} \begin{vmatrix} x_{1}^{(2)} & x_{1}^{(4)} \\ x_{2}^{(2)} & x_{2}^{(4)} \end{vmatrix}}{\begin{vmatrix} x_{1}^{(3)} & x_{1}^{(2)} \\ x_{2}^{(3)} & x_{2}^{(2)} \end{vmatrix} \begin{vmatrix} x_{1}^{(4)} & x_{1}^{(1)} \\ x_{2}^{(4)} & x_{2}^{(1)} \end{vmatrix}}.$$
 (2)

where $(A_i A_j)$ is the oriented length of the segment from the point A_i to the point A_j .

The four points can be considered in 4! = 24 different orderings which yield only six different cross ratio values (e.g., Springer, 1965, Sec. 1.8).

$$\lambda_1 = \lambda, \qquad \lambda_2 = \frac{1}{\lambda}, \qquad \lambda_3 = \frac{\lambda - 1}{\lambda},$$

$$\lambda_4 = \frac{\lambda}{\lambda - 1}, \quad \lambda_5 = \frac{1}{1 - \lambda}, \quad \lambda_6 = 1 - \lambda.$$
(3)

Any symmetric polynomial in the λ_i is already invariant of the permutation of the indices. These polynomials are also p^2 -invariants by the projective invariance of the cross ratios. The simplest one, $\sum_{i=1}^{6} \lambda_i$, is trivial having the value 3. Nontrivial permutation invariants can be obtained by considering a second degree symmetric polynomial built with the pairwise products $\lambda_i \lambda_j$.

For a systematic investigation of all the possible second-order permutation invariants, representations of the permutation group S_4 acting on the four points must be considered. It was shown in (Lenz and Meer, 1994) that these invariants can be expressed as a linear combination of

$$J_{1}[\lambda] = \frac{\lambda^{6} - 3\lambda^{5} + 3\lambda^{4} - \lambda^{3} + 3\lambda^{2} - 3\lambda + 1}{\lambda^{2}(\lambda - 1)^{2}}$$
$$J_{2}[\lambda] = \frac{2\lambda^{6} - 6\lambda^{5} + 9\lambda^{4} - 8\lambda^{3} + 9\lambda^{2} - 6\lambda + 2}{\lambda^{2}(\lambda - 1)^{2}} \quad (4)$$
$$J_{3}[\lambda] = 3$$
$$J_{4}[\lambda] = -3.$$

The nontrivial p^2 -invariants $J_1[\lambda]$ and $J_2[\lambda]$ are unbounded functions. Their ratio

$$J[\lambda] = \frac{J_2[\lambda]}{J_1[\lambda]},\tag{5}$$

however, is bounded between 2 and 2.8 and for computational convenience it is taken as the p^2 -invariant function of the four collinear points. Thus, independent of how the labels of the four points were chosen, when the cross ratio computed from (2) is used as argument in (5), the same value is obtained.

The cross ratio is very sensitive to changes in the location of the points on the line. Since the sensitivity is configuration dependent, the underlying probability distributions are complex. See (Åström and Morin, 1995; Maybank, 1994, 1995) for theoretical analyses in the context of the indexing problem. The noise sensitivity is transferred into the p^2 -invariant.

3. p^2 -Invariant in the *n*-Dimensional Projective Space

For a more detailed introduction on projective spaces see (Faugeras, 1993, Sec. 2). A point in the *n*-dimensional projective space, \mathbf{P}^n , has the homogeneous coordinates $\mathbf{x}_i = (x_1^{(i)}, \ldots, x_{n+1}^{(i)})^t$, with at least one $x_j^{(i)}$ nonzero. The standard projective basis is defined as

$$\mathbf{e}_i = (0, \dots, 1, \dots, 0)^t$$
 $i = 1, \dots, (n+1)$ (6)

$$\mathbf{e}_{n+2} = (1, 1, \dots, 1)^t \tag{7}$$

It can be shown that any (n + 2)-tuple of points in general position, i.e., no n + 1 of them are in the same hyperplane, can be transformed by a nonsingular $(n + 1) \times (n + 1)$ matrix (a projective transformation in \mathbf{P}^n) into the standard basis.

The projective invariants in \mathbf{P}^n are defined based on n+3 points in general position. Similar to (Carlsson, 1994) the notation

$$[\mathbf{x}_{1}, \dots, \mathbf{x}_{n+1}] = \begin{vmatrix} x_{1}^{(1)} & \dots & x_{1}^{(n+1)} \\ x_{2}^{(1)} & \dots & x_{2}^{(n+1)} \\ \vdots & \vdots & \vdots \\ x_{n+1}^{(1)} & \dots & x_{n+1}^{(n+1)} \end{vmatrix}$$
(8)

will be frequently used. The determinant (8) of the homogeneous coordinates of n + 1 points is proportional to the oriented volume of the parallelepiped defined by those points. In double algebra, a tool recently introduced in computer vision (Carlsson, 1994), the determinant is known as the *bracket* over the (n + 1)-dimensional vector space.

The definition of a projective invariant in \mathbf{P}^n can then be written as

$$I_{n}(\mathbf{x}_{1},...,\mathbf{x}_{n+3}) = \frac{[\mathbf{x}_{1},...,\mathbf{x}_{n-1},\mathbf{x}_{n},\mathbf{x}_{n+2}][\mathbf{x}_{1},...,\mathbf{x}_{n-1},\mathbf{x}_{n+1},\mathbf{x}_{n+3}]}{[\mathbf{x}_{1},...,\mathbf{x}_{n-1},\mathbf{x}_{n},\mathbf{x}_{n+3}][\mathbf{x}_{1},...,\mathbf{x}_{n-1},\mathbf{x}_{n+1},\mathbf{x}_{n+2}]}$$
(9)

Based on the similarity to (2) the invariant is often referred to as a cross ratio in \mathbf{P}^n . From the properties of the bracket follows that $I_n(\mathbf{x}_1, \ldots, \mathbf{x}_{n+3})$ is invariant to projective transformations of the point set, e.g., (Brill and Barrett, 1983). Thus the first n + 2 points can always be chosen as the standard projective basis. After the transformation of the point set into the standard projective basis, let the homogeneous coordinates of the last point be

$$\mathbf{x}_{n+3} = (\lambda_1, \dots, \lambda_n, 1)^t \,. \tag{10}$$

To obtain (10) the equivalence class property of homogeneous coordinates was used. For the ease of exposition the last component of the vector is taken to be nonzero, i.e., the point \mathbf{x}_{n+3} is not in the hyperplane at infinity. Extension to the general case is immediate.

The value of $I_n(\mathbf{e}_1, \ldots, \mathbf{e}_{n+2}, \mathbf{x}_{n+3})$ (note the use of standard basis) is easy to compute. We have

$$[\mathbf{e}_{1}, \dots, \mathbf{e}_{n-1}, \mathbf{e}_{n}, \mathbf{e}_{n+2}] = 1$$

$$[\mathbf{e}_{1}, \dots, \mathbf{e}_{n-1}, \mathbf{e}_{n+1}, \mathbf{x}_{n+3}] = -\lambda_{n}$$

$$[\mathbf{e}_{1}, \dots, \mathbf{e}_{n-1}, \mathbf{e}_{n}, \mathbf{x}_{n+3}] = 1$$

$$[\mathbf{e}_{1}, \dots, \mathbf{e}_{n-1}, \mathbf{e}_{n+1}, \mathbf{e}_{n+2}] = -1,$$

(11)

yielding

$$I_n(\mathbf{x}_1,\ldots,\mathbf{x}_{n+3})=\lambda_n.$$
(12)

Being a projective invariant, λ_n is called a projective coordinate of the point \mathbf{x}_{n+3} . The other λ_i can be obtained by interchanging \mathbf{e}_i and \mathbf{e}_n , i = 1, ..., (n - 1) in (9). Thus, an n + 3 point configuration in general position in \mathbf{P}^n has only n independent projective invariants.

There are (n + 3)! different labelings of the n + 3 points, and for each labeling the invariant (9) can be computed. Our goal is to derive the p^2 -invariant of the configuration. An important observation resulting from the employed definition of the invariant (9) is that $I_n(\mathbf{e}_1, \ldots, \mathbf{e}_{n+2}, \mathbf{x}_{n+3})$ remains unchanged under the permutation of the first n - 1 points, $\{\mathbf{e}_1, \ldots, \mathbf{e}_{n-1}\}$. Indeed, these permutations yield the same sign change in all four determinants which then cancel each other. The geometric interpretation is also easy to establish. The first n - 1 points define the same (n - 2)-dimensional hyperplane, \mathbf{Q} , independent of the order of the points.

To examine the influence of the permutation of the last four points, { \mathbf{e}_n , \mathbf{e}_{n+1} , \mathbf{e}_{n+2} , \mathbf{x}_{n+3} }, four (n - 1)-dimensional hyperplanes are defined together with the first n - 1 points. The hyperplanes (marked with the index of the last point) are represented by the

homogeneous coordinates

$$\mathbf{P}_{n} = (0, \dots, 0, 1)^{t}$$

$$\mathbf{P}_{n+1} = (0, \dots, 1, 0)^{t}$$

$$\mathbf{P}_{n+2} = (0, \dots, 1, -1)^{t}$$

$$\mathbf{P}_{n+3} = (0, \dots, 1, -\lambda_{n})^{t}.$$

(13)

The four hyperplanes $\mathbf{P}_n, \ldots, \mathbf{P}_{n+3}$ intersect in \mathbf{Q} , i.e., they form a pencil of planes (Faugeras, 1993, p. 26). The pair of points $\{\mathbf{e}_n, \mathbf{e}_{n+1}\}$ define the line

$$\mathbf{x} = \mu_1 \mathbf{e}_n + \mu_2 \mathbf{e}_{n+1} \,, \tag{14}$$

where (μ_1, μ_2) are the homogeneous coordinates of a point on the line in the projective basis $\{\mathbf{e}_n, \mathbf{e}_{n+1}\}$. The four intersection points of the line with the pencil of four hyperplanes, \mathbf{y}_i , have the homogeneous coordinates

$$\mathbf{y}_{n} = (1, 0)^{t}$$

$$\mathbf{y}_{n+1} = (0, 1)^{t}$$

$$\mathbf{y}_{n+2} = (1, 1)^{t}$$

$$\mathbf{y}_{n+3} = (\lambda_{n}, 1)^{t}.$$

(15)

Considering the points in the above order the value of their cross ratio (2) is $\lambda_n = I_1(\mathbf{y}_n, \dots, \mathbf{y}_{n+3})$. The definition of the line, however, was arbitrary. The projective structure of the pencil of planes assures that the effect of using a different point pair when defining the intersecting line (14) is the same on the cross ratio as a change in the order of the points \mathbf{y}_i .

Thus the effect of permutation of the last four points in the expression of the *n*-dimensional projective invariant (9) can be reduced to a permutation of the four intersection points. The 24 possible permutations yield one of the six expressions (3), and it was shown in Section 2 that the permutation dependence is eliminated by using the p^2 -invariant function $J[\cdot]$ (5). The result that the *n*-dimensional projective invariant (9) can be reduced to a one-dimensional cross ratio (2) is known for the projective plane \mathbf{P}^2 as a form of Pappus' theorem (Springer, 1965, Sec. 5.6), and is also often used in projective reconstructions, (e.g., Faugeras, 1995; Rothwell et al., 1995). The novelty of our analysis lies in exploiting it to achieve the desired permutation invariance properties. Given a labeling of the n + 3 points in \mathbf{P}^n it was therefore proven that

$$J[I_n(\mathbf{x}_1, \dots, \mathbf{x}_{n+3})] = J[I_n(\pi\{\mathbf{x}_1, \dots, \mathbf{x}_{n-1}\}, \sigma\{\mathbf{x}_n, \dots, \mathbf{x}_{n+3}\})], \quad (16)$$

for all permutations π over the first n-1 points, and all permutations σ over the last four points. Any exchange of a point \mathbf{x}_i , i = 1, ..., (n - 1) with a point $\mathbf{x}_i, j = n, \dots, (n+3)$ violates (16) since the hyperplane defined by the new first n - 1 points is different. There are only $\binom{n+3}{n-1} = \binom{n+3}{4}$ different arrangements of the labels in (16). Thus the p^2 -invariant of n + 3points in \mathbf{P}^n is a vector \mathbf{J} with $\binom{n+3}{4}$ components, each having a different argument of $J[\cdot]$. These arguments can be computed with any symbolic mathematics package (we have used MAPLE) using the standard projective basis instead of \mathbf{x}_i , i = 1, ..., (n + 2), and the projective coordinates λ_i , j = 1, ..., n for the last point. To remove the dependence on the initial labelong of the n + 3 points, the components of **J** must be ordered in (say) increasing sequence. It must be emphasized that the p^2 -invariant vector has only *n* independent components.

For four collinear points we use the p^2 -invariant function $J[\lambda]$. The p^2 -invariant of five coplanar points is a five-dimensional vector with two independent components. The p^2 -invariant of six points in the threedimensional projective space is a fifteen-dimensional vector with three independent components, etc. The derived p^2 -invariants can also be used for the dual elements (lines in \mathbf{P}^2 and planes in \mathbf{P}^3). In the following sections practical issues in the context of some applications of interest are discussed.

4. Robust Correspondence of Planar Features

In \mathbf{P}^2 , five points with homogeneous coordinates \mathbf{x}_i , i = 1, ..., 5, no three of them collinear, have two independent projective invariants which are defined as

$$\lambda_1 = I_2(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_5), \lambda_2 = I_2(\mathbf{x}_2, \mathbf{x}_1, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_5).$$
(17)

Since n = 2, the second invariant is obtained by exchanging \mathbf{x}_1 with \mathbf{x}_2 , in accordance with (16). The computed components of the p^2 -invariant **J** vector

are

$$J^{(1)} = J[\lambda_1] \quad J^{(2)} = J[\lambda_2] \quad J^{(3)} = J\left[\frac{\lambda_1}{\lambda_2}\right]$$

$$J^{(4)} = J\left[\frac{\lambda_2 - 1}{\lambda_1 - 1}\right] \quad J^{(5)} = J\left[\frac{\lambda_1(\lambda_2 - 1)}{\lambda_2(\lambda_1 - 1)}\right]$$
(18)

and are sorted in ascending order, i.e., $\mathbf{J} = (J^{[1]}, J^{[2]}, J^{[3]}, J^{[4]}, J^{[5]})^t$.

The last three $J^{(i)}$ values correspond to first exchanging \mathbf{x}_1 with \mathbf{x}_3 , \mathbf{x}_4 and \mathbf{x}_5 respectively, and substituting the computed projective invariant into the function $J[\cdot]$. Thus every point is uniquely associated with a component $J^{(i)}$ of the p^2 -invariant vector. The fact the number of points in a configuration is the same as the dimension of the \mathbf{J} vector is true only in \mathbf{P}^2 . For n > 2 additional processing must be introduced to obtain a one-to-one mapping between the components of the \mathbf{J} vector and the n + 3 points. This is discussed in Section 5.

Given a set of coplanar points, their labels are chosen arbitrarily. For each five-tuple in general position the vector **J** is computed. If a five-tuple of points from the reference set and one from the transformed set have p^2 -invariant vectors with identical components, the two five-tuples are matched. Moreover, since the components are associated with individual points, the projective correspondence of the points is also established. That is, the two five-tuples are the same up to a planar projective transformation. The projective transformation introduces eight degrees of freedom which can yield to a significant probability of mismatch, especially in the presence of outliers (points with no correspondent). Mismatches can also appear due to the high sensitivity of the invariants to positional errors. These errors are inherent since even ideal data is quantized to the closest grid location. The influence of the noise sensitivity of the invariant representations on the performance of algorithms employing them was studied mostly in the context of the indexing problem (Grimson et al., 1994; Lamdan and Wolfson, 1991; Maybank, 1995).

The performance of any algorithm processing noisecorrupted data is characterized by two probabilities: the probability of detection, and that of false alarms. These two measures are intimately connected; increasing the probability of detection (through manipulation of thresholds) will inherently increase the number of false alarms. A satisfactory probability of detection in invariance-based matching of point-sets implies a very high probability of false alarms. The ultimate goal of an efficient correspondence algorithm is to reduce the false alarm rate while minimally decreasing the probability of detection. This can only be achieved if the candidates are validated by using constraints which are *independent* of the information used in their selection process. Since the five components of the p^2 -invariant vector of a five-tuple of points are in one-to-one correspondence with the points, such constraints are easy to define, and a less sensitive correspondence algorithm can be designed.

Several procedures have to be incorporated into a robust point correspondence algorithm.

- Elimination of five-tuples with degenerate configurations (Section 4.1).
- Definition of configuration dependent bounds on the components of the *p*²-invariant vector to compensate for the effect of positional errors (Section 4.2).
- Reducing the influence on the overall performance of the correspondence algorithm of erroneously matched five-tuple pairs. This is achieved in two steps (Section 4.3) and (Section 4.4).
- Validation of the extracted point correspondences independent of the context, e.g., information about the content of the image (Section 4.5).

More details about the first three procedures can be found in (Meer et al., 1994; Ramakrishna, 1994).

4.1. Collinearity Verification

Without loss of generality we can assume in the sequel that the points have affine coordinates, i.e., $\mathbf{x}_i = (x_1^{(i)}, x_2^{(i)}, 1)$. The points in a five-tuple must be in a general position with no three of them collinear. Should three points be quasi-collinear, the area of the defined triangle, the bracket (8) is close to zero and the computation of the invariants (17) becomes numerically unstable. Let $\mathbf{x}_1, \mathbf{x}_2$ and \mathbf{x}_3 be three points. The value of their bracket is not a reliable indicator for near singularity (Golub and van Loan, 1989, p. 81), i.e., of quasi-collinearity. Instead, following (Kanatani, 1991) we define the moment matrix \mathbf{M}_{123}

$$\mathbf{M}_{123} = \sum_{i=1}^{3} x_i x_i^t.$$
 (19)

The matrix is also known as scatter or Gram matrix in the literature. The smallest eigenvalue of M_{123} is also the smallest squared singular value of the matrix $\mathbf{X} = (\mathbf{x}_1 \ \mathbf{x}_2 \ \mathbf{x}_3)$, i.e., the matrix having as columns the vectors of the three points (Biglieri and Yao, 1989). The closeness to rank deficiency of \mathbf{X} (the closeness of its smallest singular value to zero) measures how collinear the three points are (the linear dependency of their vectors).

The correspondence algorithm will not include three points into a five-tuple if the smallest eigenvalue is less than 0.001. The influence of this threshold is very weak, similar performance of the correspondence algorithm is obtained within several orders of magnitude (Ramakrishna, 1994). This is not unexpected since a threshold derived from the singular values of **X** would be considerably larger. The collinearity verification procedure is applied to both the reference and transformed sets. While collinearity is a projective invariant property, quasi-collinearity which involves metric relations, is not. However, the correspondence algorithm has a probabilistic nature and it is not necessary to use all the possible five-tuple pairs from the two sets.

4.2. Positional Uncertainty

The behavior of the one-dimensional cross ratio under i.i.d. Gaussian perturbation of the point coordinates was investigated by Maybank (1994). Åström and Morin (1995) studied the probability distribution of the cross ratio, and obtained the region of confidence for a given perturbation. Morin (1993) extended the distribution to the two-dimensional case based on experimental evidence. In a practical point correspondence algorithm, configuration dependent bounds must be established for each component of the p^2 -invariant vector, **J**, to reduce the amount of mismatched fivetuples. While linear approximations of the expressions (18) can be used, satisfactory bounds can already be obtained by the "worst case" approach.

The positional uncertainty in measuring the location of a point \mathbf{x}_i is represented as two uniformly distributed, zero-mean random variables ϵ_1^i and ϵ_2^i

$$X_1^{(i)} = x_1^{(i)} + \epsilon_1^{(i)}, \quad X_2^{(i)} = x_2^{(i)} + \epsilon_2^{(i)},$$
 (20)

where the corrupted location is $\mathbf{X}_i = (X_1^{(i)}, X_2^{(i)}, 1)$. Let ϵ be the largest half-range of $\epsilon_j^{(i)}$, j = 1, 2, over the point-set, i.e., no coordinate was corrupted beyond the interval $[-\epsilon, \epsilon]$. Neglecting the second-order error terms we have from (8)

$$[\mathbf{X}_{1}, \mathbf{X}_{2}, \mathbf{X}_{3}] = [\mathbf{x}_{1}, \mathbf{x}_{2}, \mathbf{x}_{3}] + \epsilon \sum_{j=1}^{2} \sum_{i=1}^{3} (x_{j}^{(i)} - x_{j}^{(k)}), \quad (21)$$

where k = mod(i, 3) + 1. Thus, the change in the value of a bracket due to positional error ϵ is bounded by the quantity

$$e_{123} = \epsilon \sum_{j=1}^{2} \sum_{i=1}^{3} |x_j^{(i)} - x_j^{(k)}|.$$
 (22)

The configuration-dependent minimum and maximum values are obtained by adding and subtracting e_{123} to the original value. Using the bounds for all the employed brackets, the bounds $(\lambda_1^{\min}, \lambda_1^{\max})$ and $(\lambda_2^{\min}, \lambda_2^{\max})$ for a positional uncertainty ϵ can be computed by interval arithmetic (Alefeld and Herzberger, 1983). The bounds on the other three arguments required for the p^2 -invariant vector (18) can be derived from that of λ_1 and λ_2 .

The function $J[\cdot]$ is strictly monotonic on the six intervals $(-\infty, -1)$, (-1, 0), (0, 0.5), (0.5, 1), (1, 2), $(2, \infty)$ which correspond to the six distinct cases of the one-dimensional cross ratio under permutation of the indices (3). Whenever the interval $(\lambda^{\min}, \lambda^{\max})$ is not entirely contained within one of these six intervals, using the $J[\cdot]$ function the interval can be mapped into one in which λ , the unperturbed value lies. For example, if $\lambda_1 = 0.3$, the interval $(\lambda_1^{\min}, \lambda_1^{\max})$ is mapped into (0, 0.5). Thus if $\lambda_1^{\min} = 0.1$ and $\lambda_1^{\max} = 0.6$, the bounds are set as (0.1, 0.5). The procedure is applied to all five arguments in (18).

The function $J[\cdot]$ being strictly monotonic on each interval, its bounds can be derived from the bounds of the argument. Thus, for each component of the p^2 -invariant vector $J^{[i]}$, an interval $(J^{[i],\min}, J^{[i],\max})$ is obtained. Note that the intervals established for the different components can overlap. See (Ramakrishna, 1994) for an example. Given a five-tuple of points, the three vectors \mathbf{J}^{\min} , \mathbf{J} and \mathbf{J}^{\max} determine a configuration-dependent confidence region in the five-dimensional space of the p^2 -invariant representations.

To account for positional errors when matching fivetuples, a positional uncertainty ϵ is assumed for (say) the reference set. Since projective transformations do not preserve metric properties, ϵ cannot be directly equated with a distance in the transformed set. Whenever a \mathbf{J} vector from the transformed set falls within the confidence region associated with a fivetuple from the reference set, the two five-tuples are candidates for a match. The vectors should match component-wise, i.e.,

$$J_{\text{ref}}^{[i],\min} < J_{\text{trans}}^{[i]} < J_{\text{ref}}^{[i],\max}$$
 $i = 1, \dots, 5,$ (23)

thus eliminating the problem of overlapping bounding intervals. Since the sorted component $J_{ref}^{[i]}$ is associated with the point labeled *j* in the reference set five-tuple, and $J_{trans}^{[i]}$ is associated with the point labeled *k* in the transformed set five-tuple, a matched five-tuple pair provides five point correspondences.

The main role of the parameter ϵ , is to control the number of matching candidates. Many of these candidates are incorrect due to the projective equivalence of the five-tuples and the noise sensitivity of the p^2 -invariant. It is the role of the subsequent computational modules to eliminate the erroneous correspondences. If ϵ is too small, the number of matching candidates is not large enough to extract sufficient evidence for a point correspondence. Otherwise, the performance of the algorithm remains very similar up to values of ϵ yielding the largest possible range of bounds (2, 2.8) for the **J** vector components (Ramakrishna, 1994), in which case the matching is no longer selective. The implementation uses $\epsilon = 0.4$.

4.3. Convex Hull Constraints

A perspective transformation preserves the convex hull of a point-set (e.g., Haralick and Shapiro, 1993, Sec. 13.3.3); Hartley (1993) extended the result for the projective transformation which exist between two images of the same coplanar point set. A pair of matched five-tuples puts five points in correspondence. A necessary (but not sufficient) condition for the match to be correct is that the two convex hulls should also be in correspondence.

The convex hull of five points may contain 3, 4, or 5 of the points. Thus, the following conditions can be defined.

- 1. The number of points on the convex hull must be the same.
- 2. Corresponding points must lie either on or inside the convex hull.
- 3. For points lying on the convex hull, neighborhood relations must be preserved.

The last condition is meaningful only if the convex hull contains at least four points, in which case the correspondence between two points implies the correspondences between their neighbors as well.

The convex hull preservation was employed by Morin (1993) to construct two p^2 -type invariants for a five-tuple of coplanar points. Our p^2 -invariant representation did not use the convex hull of the five-tuple, and therefore the convex hull constraints provide the most effective *independent* information for the validation of the matching candidates. Note that the constraints imply that the point correspondences are hypothesised between the matched five-tuples. As was discussed in the previous section this is available from the p^2 -invariant representation.

Should more than one matching candidate remain after the verification of the convex hull constraints, the candidate with the smallest J_{ref} , J_{trans} distance in the five-dimensional space of the p^2 -representations, computed with L_2 norm is chosen. Our experiments with synthetic data have shown that for a noisy transformed set the correct match is often not the closest one among all the candidates. Thus, after applying the convex hull constraints it is not guaranteed that the nearest-neighbor rule selects the correct candidate. The consensus-seeking technique described in the next section, can tolerate a significant amount of mismatches. Retaining all the candidates which verified the convex hull constraint, will introduce more mismatches than when keeping only one candidate. The error tolerance of the correspondence algorithm is well illustrated by the fact that the performance does not change significantly in former case.

4.4. Tolerated Mismatch

The point correspondences obtained from a matched pair of five-tuples can contain zero to five errors. Let the probability of an incorrectly detected point correspondence be P_{error} . Extensive simulations have shown that P_{error} remains the same for all the points in a five-tuple.

Assume for convenience that the two sets contain the same number of points, *N*, and that all the points have an equal chance to pass the matching criterion. Every established correspondence casts a vote in a *contingency table*. The contingency table has the labels of the reference set as row addresses and the labels of the transformed set as column addresses. Thus a reference and a transformed point pair uniquely defines a cell.

A reference/transformed point correspondence casts a vote into the correctly addressed cell with probability $1 - P_{\text{error}}$. The same reference point will cast, under the uniform sampling assumption, incorrect votes into a given cell with probability $P_{\text{error}}/(N-1)$. To be able to detect the correct point correspondence after all the votes were cast, the latter probability should not exceed the former.

The contingency table is an effective tool for accumulating the votes since it distributes the incorrect ones over the entire set of points. Indeed, the above condition gives the tolerated probability of error in pairing a point from the reference set with a point from the transformed set:

$$P_{\rm error} < 1 - \frac{1}{N}.$$
 (24)

The relation (24) predicts that P_{error} can exceed 0.8, with most of the point pairs erroneously voting into the contingency table while the two sets are still put into correct correspondence. Experimental results have shown that this is indeed the case for $N \ge 10$.

When a sufficiently large number of transformed set five-tuples were matched, enough votes were entered into the contingency table. To extract the point correspondences, a simple greedy algorithm is used. The cell with the highest number of votes in the table identifies the first correspondence. This pair of points is eliminated by removing from the table the row and the column of the cell. The procedure is then repeated until all the row (or column) entries were used. Note that it is not necessary to have the N correct point correspondences associated with the N largest values in the initial contingency table. It suffices that at every iteration the largest value in the table indicates a correct correspondence. This property of the algorithm compensates for the nonuniform point selection almost always present in real data, i.e., some points are more frequently present in the matched five-tuples than others.

For a set of N points $O[N^5]$ different five-tuples can be defined, and thus the same number of p^2 -invariant representations. Without label permutation invariance, the number of projective representations increases more than two orders of magnitude (5! = 120). For the reference set all the p^2 -invariant representations have to be computed, however, to generate enough votes in the contingency table only a subset of the possible transformed set five-tuples has to be processed. These five-tuples can be chosen by random sampling, and their number depends on the amount of positional error present in the data. For slightly corrupted data 20–30 percent of the total number is enough.

The contingency table is a computational tool through which a consensus decision is extracted from the mostly incorrect point correspondences. (In experiments with synthetic data and positional errors of a few pixels, Perror often exceeded 0.8!). As discussed above, the success of the technique is due to the distribution of errors across the entire table. The contingency table based voting technique is basically different from the accumulation procedure used in geometric hashing (Lamdan et al., 1990). An analytical comparison of error sensitivities is beyond the goal of this paper, but it is well-known that geometric hashing has a rather poor performance once the transformation group becomes more complex than rigid motion (Grimson et al., 1994; Lamdan and Wolfson, 1991). The size of the hashing table required for accumulating the votes in the projective case is $O[N^4]$ with a constant 4! = 24, while the size of the contingecy table is $N \times N$.

4.5. Correspondence Validation

The output of the contingency table analysis is the correspondence between the points in the reference and the transformed sets. These correspondences must be validated since if the two sets do not share all the points, i.e., outliers are present, false correspondences are introduced. An efficient validation technique should also remove most of the correspondence errors and recognize when the algorithm failed completely. The natural way for validation is backprojection of the transformed set onto the reference set and computation of a quality measure. Exploiting the fact that the correspondences were already established, a consensus-seeking method similar to the contingency table based analysis can be used.

A planar projective transformation is uniquely determined by four point correspondences. Thus if N point correspondences were established, $\binom{N}{4}$ possible transformations can be defined. Note the relative small number, in contrast to the number needed for the alignment method (Huttenlocher and Ullman, 1990). The N correspondences are labeled from 1 to N and associated with empty accumulator cells. Let a transformation be obtained from the correspondences labeled 1 to 4. The remaining N - 4 points of the transformed set are projected into the reference set. For these points, d_i^2 the squared Euclidean distance to the corresponding reference point is computed, and sorted in ascending order $d_{[i]}^2$. A robust quality measure of the backprojection can be defined as

$$\delta_{1234} = \frac{1}{\sum_{i=m_1}^{m_2} d_{[i]}^2},\tag{25}$$

where $m_1 = 1$ and $m_2 = \lfloor 0.4 * (N - 4) \rfloor$. The indices m_1 and m_2 control the sensitivity of the validation procedure. The trade-off is between exclusion of large distances due to errors (m_2 small), and increased support for the computation of δ_{1234} (m_2 large). The accumulator cells 1 to 4 are incremented with δ_{1234} . After all (or a large number) of the possible transformations were used, the confidence in a point correspondence is given by the value in the associated accumulator cell.

Whenever an incorrect point correspondence is used to compute the transformation, δ is more likely to decrease. The cumulative effect keeps the value of those accumulator cell which are associated with such correspondences significantly lower. In the absence of large positional uncertainties outliers are immediately revealed having confidences an order of magnitude smaller.

Large positional uncertainties level off all the confidence values and therefore it is difficult to establish a unique validation threshold. Data-dependent thresholds, however, can be derived from *N* random correspondences between the two sets. The above described confidence computation procedure is performed with these random pairings of points. To compensate for accidental proximity, the summation limits in (25) are increased to $m_1 = 3$, and m_2 modified accordingly. For large number of points $m_1 = 2$ can also be used.

All the "random" confidence values belong to the same (unknown) distribution. The mean and the standard deviation of the distribution are computed and the validation threshold is set as the mean plus three standard deviations. The accuracy of this threshold does not seem to improve if the distribution is based on more than one random shuffling of the correspondences. The method employed to derive the data-dependent threshold belongs to the family of resampling techniques, more precisely to bootstrapping (Efron and Tibshirani, 1993).

A correspondence established by the algorithm is declared valid if the associated confidence value exceeds the threshold. The validated point correspondences can be used to compute by least squares (or if additional safeguards are to be taken, by the robust least median of squares) the transformation between the two point-sets. Any practical algorithm should be able to diagnose its own failure. The validation method declares complete breakdown whenever less than four point correspondences are declared valid. In such cases context-dependent information must be used.

4.6. Algorithm Summary

The processing steps are reviewed below:

- 1. For each nondegenerate five-tuple in the reference set its p^2 -invariant representation together with the bounds (for $\epsilon = 0.4$) are computed.
- 2. The p^2 -invariant representation for *K* randomly chosen nondegenerate five-tuples from the transformed set are derived, and the matching candidates established.
- 3. The best candidates (if any) are retained after pruning with the convex hull constraints.
- 4. The obtained point correspondences vote into the contingency table, from where the correspondence between the two point-sets is extracted.
- 5. The output of the algorithm is validated through backprojection.

Note that context-dependent (task-specific) information is not used at any stage of the algorithm. Should such information be available, it can provide additional safeguards against erroneous decisions.

4.7. Experimental Results

The sensitivity of the p^2 -invariant representations was studied through the performance of the correspondence algorithm for synthetic data and real images. In case of the synthetic data both the reference and the transformed sets contained the same number of points, N = 15. The projective camera model had a focal distance 50. The points to be used for the reference set were assumed to be in a plane perpendicular to the optical axis Oz at distance 300. Their coordinates were randomly chosen between (-127, -127) and (128, 128)and projected into the image plane. The optical axis pierced the reference scene and image planes at (0, 0). The image was then rescaled independently along the x and y coordinates to extend over a 256×256 pixel array, and the point coordinates were quantized to the closest pixel center.

To obtain the *transformed set*, the reference scene plane was rotated around the three axes in a random order. The closeness of the image plane put limitations on the range of allowed rotations. The maximum range

of rotations around Ox and Oy axes was $(-57^{\circ}, 57^{\circ})$, and full rotations were allowed around the optical axis. In the few cases when points in the transformed scene plane moved behind the image plane, the artifact was eliminated by iteratively reducing the amount of rotation that caused it. The transformed scene plane was then projected into the image plane and rescaled to a 256×256 array.

The *positional uncertainty* was introduced into the transformed set by independently perturbing the x_1 and x_2 coordinates of the points with uniformly distributed random variables having half-range u, and thus a maximum shift in location of $\sqrt{2u}$ could be introduced by the noise. Outliers were obtained by replacing one to three transformed set points with points at a randomly chosen location. After the perturbation, the point coordinates were quantized to the closest pixel center. Note that when deriving performance statistics no constraints on minimum interpoint distances or collinearity (beside that in Section 4.1) are used.

The performance of the correspondence algorithm mirrors the sensitivity of the p^2 -invariant representations to perturbations. For every experimental condition the performance was assessed based on 100 trials. The number of five-tuples chosen from the transformed set was K = 2000. The output of interest are the validated point correspondences, since these are the ones used in further processing, either to compute the transformation between the reference and transformed set, or to recover 3D information. Four probabilities were measured from the trials.

- $P_{bv}(k)$: The cumulative probability that before validation there are *at most k* erroneous correspondences among the extracted 15 pairs. Thus, even if an outlier was put into correspondence with an outlier, this is considered as an error since such correspondence cannot be used to recover the transformation between the sets. The cumulative probability $P_{bv}(k)$ characterizes the input into the validation module.
- *p_F*: The probability of a trial being declared "failure", i.e., less than four correspondences are validated. The probability *p_F* characterizes the self-diagnosis capability of the algorithm.
- $P_{av}(k)$: The cumulative probability that there are *at* most k erroneous correspondences retained after validation of the trials not classified as failure. The cumulative probability $P_{av}(k)$ characterizes the error detection performance of the validation module.

k	0	1	2	3	4	5	6	7	8	9	10	11	12
$P_{bv}(k)$	0.81	0.81	0.96	0.97	0.99	1							
$P_{av}(k)$	0.93	0.94	1										
$p_{fr}(k)$	0.98	0.01	0	0.01									

Number of outliers: 0. Half-range of the noise: 4 pixels.

Number of outliers: 1. Half-range of the noise: 4 pixels.

k	0	1	2	3	4	5	6	7	8	9	10	11	12
$P_{bv}(k)$	0	0.61	0.69	0.77	0.85	0.92	0.97	1					
$P_{av}(k)$	0.68	0.95	0.97										
$p_{fr}(k)$	0.93	0.02	0.01	0.01									

Number of outliers: 2. Half-range of the noise: 2 pixels.

k	0	1	2	3	4	5	6	7	8	9	10	11	12
$\begin{array}{c} P_{bv}(k) \\ P_{av}(k) \\ p_{fr}(k) \end{array}$	0 0.57 0.96	0 0.89	$\begin{array}{c} 0.68\\ 0.96\end{array}$	0.84	0.89	0.94	0.95	1					

Number of outliers: 2. Half-range of the noise: 3 pixels.

k	0	1	2	3	4	5	6	7	8	9	10	11	12
$P_{bv}(k)$	0	0	0.42	0.60	0.73	0.83	0.89	0.95	0.99	1			
$P_{av}(k)$	0.62	0.84	0.86	0.89	0.90								
$p_{fr}(k)$	0.85	0.03	0.01	0.01									

Number of outliers: 2. Half-range of the noise: 4 pixels.

k	0	1	2	3	4	5	6	7	8	9	10	11	12
$P_{bv}(k)$	0	0	0.25	0.40	0.57	0.68	0.78	0.85	0.91	0.96	0.98	0.98	1
$P_{av}(k)$	0.44	0.63	0.74	0.75									
$p_{fr}(k)$	0.71	0.03	0.01										

Number of outliers: 3. Half-range of the noise: 2 pixels.

k	0	1	2	3	4	5	6	7	8	9	10	11	12
$P_{bv}(k)$	0	0	0	0.43	0.60	0.77	0.85	0.92	0.92	0.97	0.98	0.99	0.99
$P_{av}(k)$	0.54	0.80	0.86										
$p_{fr}(k)$	0.84	0.02											

Figure 1. Experimental results for 15 points synthetic data. See text.

• $p_{fr}(k)$: The probability that there are k correct correspondences rejected during the validation of the trials not classified as failure. The probability $p_{fr}(k)$ characterizes the false alarm performance of the validation module.

The mixed use of cumulative and individual probabilities was found to be the most informative way for characterizing the performance of the correspondence algorithm. These probabilities are related

$$\sum_{k=1}^{12} p_{fr}(k) = \max_{k} P_{av}(k) = 1 - p_F.$$
 (26)

In Fig. 1 the results obtained for several experimental conditions are tabulated. The number of outliers was between zero and three, and positional uncertainty of 2 to 4 pixels were introduced for both coordinates. As expected, the performance decreases with the increase of the perturbation, either through positional uncertainty or outliers. Once the value of a cumulative probability reaches its maximum it is no longer shown. The validation module succeeds to recover most of the correct point correspondences while keeping a relative low failure rate (recall that the uncorrupted input has almost all the points in correspondence). The failure rate, p_F can easily be determined from Fig. 1 using (26). For example, if the original data contained two outliers and was corrupted with positional uncertainty of 3 pixels, $p_F = 0.1$. In (Meer et al., 1994) and (Ramakrishna, 1994) additional experimental data, characterized by a different interpretation of $P_{bv}(k)$, as well as detailed analysis of a single trial, can be found. To measure the simulated probabilities with high accuracy an unfeasible large number of trials would be required. We have found, however, that even with 100 trials the statistical variations of the measured probabilities are satisfactorily low.

Morin (1993; Morin et al., 1994), performed a large comparative study on the noise sensitivity for several p^2 -type invariant representations of a five-tuple of collinear points. The two sets contained 50 points defined on a 512 × 512 image, from which 12 five-tuples were chosen in the reference set as model candidates. The allowed configurations were selected for minimum interpoint distance (40 pixels) and noncollinearity (at least 17 degrees deviations). The introduced positional uncertainty was zero-mean Gaussian with 1 pixel standard deviation. All the invariant representations showed high sensitivity (measured through matching tasks), which probably can be attributed to the presence of many outliers.

Morin has concluded that invariant representations based on symmetric polynomials (the class to which $J[\cdot]$ also belongs) are less effective than a convex hull based representation. As was discussed in Section 4.3 it is very important to use the information provided by the convex hull for validation and not for deriving invariant representations. The performance of our correspondence algorithm (tolerating more severe positional uncertainty for unconstrained configurations) seems to confirm this observation.

The p^2 -invariant vector **J** has only two independent components, and for matching the components $J^{[1]}$ and $J^{[2]}$ should suffice. The effect of using nonredundant representations in the correspondence algorithm was somewhat unexpected, underlining the complex probabilistic processes on which the correspondence algorithm is based. The amount of required computations instead of decreasing it increased significantly due to the larger number of selected candidates. For moderate perturbations the performance of the algorithm remained practically unchanged, otherwise it declined. For example, with three outliers and a halfrange of the noise 2 pixels, $p_F = 0.26$ (instead of 0.14), $P_{av}(0) = 0.44$ (instead of 0.54), and $p_{fr}(0) = 0.70$ (instead of 0.84). We conclude, similar to Morin (1993) and Rothwell et al. (1995), that using redundant invariant representations is necessary to compensate for the noise sensitivity of these representations.

In Fig. 2 the performance of the point correspondence algorithm for a real image is shown. The reference and the transformed sets contain N = 20 points each, extracted from the images Fig. 2(a) and (b) respectively. The integer coordinates of the points were extracted manually with a cursor and thus positional errors are certainly present. Note that the coplanarity condition of the points is only approximately satisfied, and many quasicollinear configurations can be defined. The last four points (labeled 17 through 20) are outliers being taken as different in the two images.

The algorithm was run on the data with K = 10000. Only nine of the extracted correspondences are correct. The correct correspondences are associated with the largest confidences, and they are all validated (Fig. 2(c)). Note the relative small range of the confidence values, and the effectiveness of the validation threshold derived from the data. The transformation between the images can now be computed and the two images aligned. The registration of the same pair of images using an algorithm which requires a priori information (Zheng and Chellappa, 1992), is described in (Chellappa et al., 1993).

5. P²-Invariant Representation in P³

In \mathbf{P}^3 , six points with homogeneous coordinates \mathbf{x}_i , i = 1, ..., 6, no four of them coplanar, have three independent invariants which can be taken as

$$\lambda_{1} = I_{3}(\mathbf{x}_{3}, \mathbf{x}_{2}, \mathbf{x}_{1}, \mathbf{x}_{4}, \mathbf{x}_{5}, \mathbf{x}_{6})$$

$$\lambda_{2} = I_{3}(\mathbf{x}_{1}, \mathbf{x}_{3}, \mathbf{x}_{2}, \mathbf{x}_{4}, \mathbf{x}_{5}, \mathbf{x}_{6})$$

$$\lambda_{3} = I_{3}(\mathbf{x}_{1}, \mathbf{x}_{2}, \mathbf{x}_{3}, \mathbf{x}_{4}, \mathbf{x}_{5}, \mathbf{x}_{6}).$$
(27)

The definition is arbitrary, any three arrangement satisfying (16) can be used. The p^2 -invariant vector **J** has fifteen components which must be sorted in ascending order.

$$J^{(1)} = J[\lambda_1] \quad J^{(2)} = J[\lambda_2]$$
$$J^{(3)} = J\left[\frac{\lambda_1}{\lambda_2}\right] \quad J^{(4)} = J\left[\frac{\lambda_2 - 1}{\lambda_1 - 1}\right]$$
$$J^{(5)} = J\left[\frac{\lambda_1(\lambda_2 - 1)}{\lambda_2(\lambda_1 - 1)}\right] \quad J^{(6)} = J[\lambda_3]$$



R	3	18	11	6	15	20	4	10	9	19
Т	10	3	18	20	4	9	11	15	17	6
Conf. Valid	2.006 n	2.025 n	2.040 n	2.042 n	2.056 n	2.113 n	2.167 n	2.213 n	2.509 n	2.535 n
R	17	7	2	14	5	16	8	1	13	12
Т	19	7	2	14	5	16	8	1	13	12
Conf. Valid	2.600 n	3.747 у	4.292 y	4.455 y	5.132 y	5.154 y	5.233 y	5.285 y	5.373 y	5.527 y
					(c)					

Figure 2. Correspondence for real data: (a) Reference image. (b) Transformed image. The points labeled 17 to 20 are outliers. (c) Extracted correspondences and confidence validation.

$$J^{(7)} = J \begin{bmatrix} \frac{\lambda_1}{\lambda_3} \end{bmatrix} \quad J^{(8)} = J \begin{bmatrix} \frac{\lambda_3 - 1}{\lambda_1 - 1} \end{bmatrix}$$
$$J^{(9)} = J \begin{bmatrix} \frac{\lambda_1(\lambda_3 - 1)}{\lambda_3(\lambda_1 - 1)} \end{bmatrix} \quad J^{(10)} = J \begin{bmatrix} \frac{\lambda_2}{\lambda_3} \end{bmatrix}$$
$$J^{(11)} = J \begin{bmatrix} \frac{\lambda_3 - 1}{\lambda_2 - 1} \end{bmatrix} \quad J^{(12)} = J \begin{bmatrix} \frac{\lambda_2(\lambda_3 - 1)}{\lambda_3(\lambda_2 - 1)} \end{bmatrix}$$
$$J^{(13)} = J \begin{bmatrix} \frac{\lambda_2 - \lambda_1}{\lambda_3 - \lambda_1} \end{bmatrix} \quad J^{(14)} = J \begin{bmatrix} \frac{\lambda_3(\lambda_2 - \lambda_1)}{\lambda_2(\lambda_3 - \lambda_1)} \end{bmatrix}$$
$$J^{(15)} = J \begin{bmatrix} \frac{(\lambda_3 - 1)(\lambda_2 - \lambda_1)}{(\lambda_2 - 1)(\lambda_3 - \lambda_1)} \end{bmatrix}$$
(28)

The fifteen arguments are identical (up to a permutation of the point labels) with the invariants derived independently in (Rothwell et al., 1995) for *a given* labeling of the point set. Our result eliminates all dependencies on the ordering of the six points.

Carlsson (1995) has shown recently that the threedimensional invariant defined by five points in \mathbf{P}^3 together with the center of perspective, is the same as the two-dimensional invariant of their projection onto an arbitrary plane. The result is a generalization of the reducibility of an invariant defined in a higher dimension to one in a lower dimension (see Section 3), and can be used to interpret (28). From (27) results that $J^{(1)}$ to $J^{(5)}$ in (28) correspond to the two-dimensional p^2 -invariant representation (18) of the projections of the five points { \mathbf{x}_2 , \mathbf{x}_3 , \mathbf{x}_4 , \mathbf{x}_5 , \mathbf{x}_6 }. Similarly, using (16), $J^{(6)}$ to $J^{(9)}$ correspond to the p^2 -invariant representation of the five-tuple obtained with the point \mathbf{x}_2 being left out ($J[\lambda_1]$ being already generated). The components $J^{(10)}$ to $J^{(12)}$ are obtained with the point \mathbf{x}_3 discarded ($J[\lambda_1]$ and $J[\lambda_3]$ are already available). The last three components, $J^{(13)}$ to $J^{(15)}$ capture the intrinsic three-dimensional structure of the six points, and depend on all three invariants.

To build a contingency table when the projective space has dimension n > 2, the intrinsic structure of the p^2 -invariant vector has to be exploited. In \mathbf{P}^2 the components of the vector are in one-to-one correspondence with the points in the five-tuple. In \mathbf{P}^3 pairs of points correspond to the vector components. Thus $J^{(1)}$ is associated with the pair $(\mathbf{x}_1, \mathbf{x}_2), J^{(2)}$ with the pair $(\mathbf{x}_1, \mathbf{x}_3)$, $J^{(3)}$ with the pair $(\mathbf{x}_1, \mathbf{x}_4)$, and so on, $J^{(15)}$ being associated with the pair (\mathbf{x}_5 , \mathbf{x}_6). After two fifteen-dimensional vectors are matched, the correspondence between the six point pairs can be extracted with a "local" 6×6 table. The redundancy of the p^2 -invariant representation somewhat compensates for matching errors. A confidence measure in the correspondence is also obtained and weighted votes can be accumulated in the contingency table.

The problem of six points in \mathbf{P}^3 is central to projective reconstruction, i.e., using uncalibrated cameras. In this case, the projective coordinates of the six points have to be recovered from two (or more) images of the configuration. When the epipolar geometry (the fundamental matrix) of a pair of uncalibrated cameras is known, the invariants λ_1 , λ_2 , λ_3 can be computed after performing simple geometric constructions with the images of the six points (Carlsson, 1994; Rothwell et al., 1995). If three images are available, the projective coordinates can be recovered without knowing the epipolar geometry (Quan, 1995).

The estimated projective coordinates are sensitive to positional uncertainty (Rothwell et al., 1995). A projective transformation in \mathbf{P}^3 has 15 degrees of freedom, and therefore it is expected that using p^2 -invariant representations to index into data bases of 3D models (or seeking correspondence between point sets) will require additional safeguards, similar to those described in Section 4. How such, context-independent constraints (like the convex hull preservation in \mathbf{P}^2) can be built, is an open question not addressed here.

6. Discussion

Two new concepts were introduced in this paper. The class of p^2 -invariant representations are economical representations of a point-set (or its dual configuration), even when the recommended redundant representations are employed. Permutation invariants were derived for other transformation groups in (Lenz and Meer, 1994). For simple groups the effect of label permutations can also be eliminated by exploiting metric properties, e.g., volumes are invariant under rigid motion.

The second new concept introduced is the use of consensus-based techniques to increase the noise tolerance of an algorithm. From mostly erroneous point correspondence hypotheses, reliable decisions were derived since the errors were distributed across the space of all possible correspondences, while the sought evidence was accumulated in a single cell. The technique was successfully applied twice, for the contingency table and for the final validation. Consensus seeking provides robust behavior (see the real image experiment where the validation removed all the wrong correspondences, i.e., outliers). This is not unexpected since the consensus technique is based on the same principle as the robust estimators using elemental subsets, e.g., least median of squares.

We have shown that if additional constraints are used to eliminate false matches, the p^2 -invariant representations can be used to build practical algorithms either for the correspondence or the indexing problem. It is important that these constraints are context-independent, i.e., are not based on information restricted to the input image. Such information should provide only an additional layer of task-specific safeguards to enhance performance.

It is probably not very difficult to adapt an existing, invariance-based object recognition system to use p^2 -invariants, but this was never attempted in the paper. Such a system, should first break the *global* correspondence problem into several (possibly overlapping) subproblems, to keep the number of outliers relatively low. Possible approaches to delineate subregions can use salient convex groups (Jacobs, 1996), or class-driven grouping (Mundy et al., 1994b). However, the grouping has to be chosen to only minimally interfere with the definition of p^2 -invariants. Based on our experimental results, the performance should improve, but it is safe to assume that the p^2 -invariants will not be able to provide the ultimate solution for object recognition, which definitely needs a top-down component.

To conclude, it is important to recognize a fundamental question raised by the paper. For superior performance in a practical object recognition system an optimal balance must be achieved between the information used in deriving invariant representations, and the information used in validating the hypotheses derived from these (inherently) unreliable representations.

Acknowledgment

We would like to thank the three anonymous reviewers for their thorough analysis of the paper which improved the presentation. The research was started when the first two authors spent the summer of 1992 together at the Advanced Telecommunication Research Institute, Seika-Cho, Kyoto, Japan, and they are grateful for the opportunity. The implementation of the validation procedure was done by Thomas Slowe. Peter Meer and Sudhir Ramakrishna were supported by the National Science Foundation grant IRI-9210861, Reiner Lenz under TFR Grant 93-192.

References

- Alefeld, G. and Herzberger, J. 1983. Introduction to Interval Computations. Academic Press: New York.
- Åström, K.E. and Morin, L. 1995. Random cross ratios. In Proc. 9th Scand. Conf. on Image Anal., Uppsala, Sweden, pp. 1053–1061.
- Biglieri, E. and Yao, K. 1989. Some properties of singular value decomposition and their applications to digital signal processing. *Signal Processing*, 18:277–289.
- Brill, M.H. and Barrett, E.A. 1983. Closed-form extension of the anharmonic ratio to *n*-space. *Computer Vision, Graphics, and Image Processing*, 23:92–98.
- Carlsson, S. 1994. The double algebra: An effective tool for computing invariants in computer vision. In *Applications of Invariance in Computer Vision*, J.L. Mundy et al. (Eds.), Springer Verlag, pp. 145–164.
- Carlsson, S. 1995. Duality of reconstruction and positioning from projective views. In *Proc. of IEEE Workshop on Representation* of Visual Scenes, Boston, pp. 85–92.
- Chellappa, R., Zheng, Q., Davis, L.S., DeMenthon, D., and Rosenfeld, A. 1993. Site-model-based change detection and image registration. In *Proc. DARPA Image Understanding Workshop*, Washington, DC, pp. 205–216.
- Efron, B. and Tibshirani, R.J. 1993. *An Introduction to the Bootstrap.* Chapman & Hall: London.
- Faugeras, O. 1993. *Three Dimensional Computer Vision*. MIT Press: Cambridge, MA.
- Faugeras, O. 1995. Stratification of three-dimensional vision: Projective, affine, and metric representations. J. Opt. Soc. Am. A, 12:465–484.

- Golub, G.H. and van Loan, C.F. 1989. *Matrix Computations*. Second edition. Johns Hopkins University Press: Baltimore.
- Grimson, W.E.L., Huttenlocher, D.P., and Jacobs, D.W. 1994. A study of affine matching with bounded sensor error. *Intl. J. of Computer Vision*, 13:7–32.
- Haralick, R.M. and Shapiro, L.G. 1993. *Computer and Robot Vision*. Addison-Wesley: Reading, MA, Vols. I–II.
- Hartley, R. 1993. Cheirality invariants. In Proc. of DARPA Image Understanding Workshop, Washington, D.C., pp. 745–753.
- Huttenlocher, D.P. and Ullman, S. 1990. Recognizing solid objects by alignment with an image. *Intl. J. of Computer Vision*, 5:195– 212.
- Jacobs, D.W. 1996. Robust and efficient detection of salient convex groups. *IEEE Trans. on PAMI*, 18:23–38.
- Kanatani, K. 1991. Computational projective geometry. CVGIP: Image Understanding, 54:333–348.
- Lamdan, Y., Schwartz, J.T., and Wolfson, H.J. 1990. Affine invariant model-based object recognition. *IEEE Trans. Robot. Autom.*, 6:578–589.
- Lamdan, Y. and Wolfson, H.J. 1991. On the error analysis of 'Geometric Hashing'. In Proc. of the 1991 IEEE Comp. Soc. Conf. on Computer Vision and Pattern Recog., Lahaina, Maui, Hawaii, pp. 22–27.
- Lenz, R. and Meer, P. 1994. Point configuration invariants under simultaneous projective and permutation transformations. *Pattern Recog.*, 27:1523–1532.
- Maybank, S.J. 1994. Classification based on the cross ratio. In *Applications of Invariance in Computer Vision*, J.L. Mundy et al. (Eds.), Springer Verlag, pp. 453–472.
- Maybank, S.J. 1995. Probabilistic analysis of the application of the cross ratio to model based vision: Misclassification. *Intl. J. of Computer Vision*, 14:199–210.
- Meer, P., Ramakrishna, S., and Lenz, R. 1994. Correspondence of coplanar features through p²-invariant representations. In *Applications of Invariance in Computer Vision*, J.L. Mundy et al. (Eds.), Springer Verlag, pp. 473–492.
- Morin, L. 1993. *Quelques Contributions des Invariants Projectifs a la Vision par Ordinateur*, Ph.D. Thesis, Institut National Polytechnique de Grenoble.
- Morin, L., Brand, P., and Mohr, R. 1994. Indexing with projective invariants. Presented at the *Syntactical and Structural Pattern Recog. Workshop*, Nahariya, Israel.
- Mundy, J.L. and Zisserman, A. (Eds.) 1992. *Geometric Invariance in Machine Vision*. MIT Press: Cambridge, MA.
- Mundy, J.L., Zisserman, A., and Forsyth, D.A. (Eds.) 1994a. Applications of Invariance in Computer Vision. Lecture Notes in Computer Science, 825. Springer-Verlag: Heilderberg.
- Mundy, J.L., Huang, C., Liu, J., Hoffman, W., Forsyth, D.A., Rothwell, C.A., Zisserman, A., Utcke, S., and Bournez, O. 1994b. MORSE: A 3D object recognition system based on geometric invariants. In *Proc. DARPA Image Understanding Workshop*, Monterey, CA, pp. 1393–1402.
- Quan, L. 1995. Invariants of six points and projective reconstruction from three uncalibrated cameras. *IEEE Trans. on PAMI*, 17:34–46.
- Ramakrishna, S. 1994. Invariance Based Robust Matching. M.S. Thesis. Tech. Rep. CE-105. Dep. of Electrical and Comp. Engn., Rutgers University.
- Rothwell, C.A. 1994. Hierarchical object description using invariants. In *Applications of Invariance in Computer Vision*, J.L. Mundy et al. (Eds.), Springer Verlag, pp. 397–414.

- Rothwell, C.A., Csurka, G., and Faugeras, O. 1995. A comparison of projective reconstruction methods for pair of views. INRIA Tech. Rep., No. 2538. Appeared in *Computer Vision and Understanding*, 68:37–58, 1997.
- Springer, C.E. 1965. *Geometry and Analysis of Projective Spaces*. Freeman: San Francisco.
- Zhang, S., Deriche, R., Faugeras, O., and Luong, Q.-T. 1994. A robust technique for matching two uncalibrated images through the recovery of unknown epipolar geometry. INRIA

Tech. Rep., No. 2273. Artificial Intelligence, 78:87–119, 1996.

- Zheng, Q. and Chellappa, R. 1992. A computational vision approach to image registration. In *Proc. 11th IAPR Intl. Conf. on Pattern Recog.*, The Hague, The Netherlands, Vol. 1, pp. 193–197.
- Zisserman, A., Mundy, J., Forsyth, D.A., Liu, J., Pillow, N., Rothwell, C.A., and Utcke, S. 1995. Class-based grouping in perspective images. In *Proc. 5th Intl. Conf. in Computer Vision*, Boston, pp. 183–188.