TR－A－0149

ERROR TOLERANT METHOD FOR INVARIANCE BASED FEATURE CORRESPONDENCE

Peter Meer

1992. 8. 19

ATR 視聴覚機構研究所
〒619-02 京都府相楽郡精華町光台 2-2 ☎07749-5-1411

ATR Auditory and Visual Perception Research Laboratories
2-2, Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-02 Japan
Telephone: +81-7749-5-1411
Facsimile: +81-7749-5-1408

ERROR TOLERANT METHOD FOR INVARIANCE BASED FEATURE CORRESPONDENCE

Peter Meer*

Cognitive Processes Department
ATR Auditory and Visual Perception Research Laboratories
2-2 Hikaridai, Seika-cho, Soraku-gun
Kyoto 619-02, Japan

ABSTRACT

A two stage probabilistic algorithm for establishing feature correspondence between images is presented. First, k-tuples of features are matched based on similar invariant representations. The representation should be invariant not only to the group of image transformations but also to the permutation group of the k elements. A projective and permutation invariant representation for 5-tuples of points/lines in a plane is described. In the second stage, the algorithm recovers feature correspondence from a contingency table built with the ensemble of matched k-tuples. The conditions for reliable performance are given, and it is shown that the correct solution for the correspondence problem can be be found while most the data in the matched k-tuples is erroneous. The algorithm does not use global information and recovers feature correspondence in $O[n^2]$ time.

Keywords: robust computer vision, invariants, feature correspondence, object recognition.

---

* On leave from the Department of Electrical and Computer Engineering, Rutgers University, Piscataway, NJ 08855-0909, USA.

1. Introduction

Establishing the correspondence between features in two image frames (or finding the instance of a model) is an important prerequisite for the execution of many computer vision tasks. The two sets of features involved in a correspondence problem will be referred to as the *reference set* and the *transformed set*. A projective transformation is assumed to map a subset of features from the reference to the transformed set. The presence of noise introduces uncertainty and for convenience all the uncertainty is allocated to the features in the transformed set. This assumption bears no relevance for the proposed method and it is true in object recognition where the reference set contains the models. Our goal is to develop a robust object recognition system which tolerates significant feature localization errors. The correspondence algorithm described in this paper is a component of the system but can also be used in stereo or motion tasks.

Feature correspondence is required (although often only implicitly) in most of the recently proposed object recognition methods. In the *alignment methods* (also known as transform determination, or clustering techniques) the affine transformation between the reference and transformed sets is sought. To compute the transformation correspondence is assumed between two configurations of noncollinear 3-tuples of points randomly chosen from the reference and transformed sets. (Equivalent configurations were also used.) To validate whether the obtained transformation can account for the data, Huttenlocher and Ullman (1990) backprojected the model. Whenever the two configurations are in correspondence the computed transformations map into nearby locations in the six-dimensional feature space of the transformation parameters. Mundy and Heller (1990) exploited this property and extracted the significant clusters from the feature space. Validation was again performed by model backprojection. In practice the alignment methods are very sensitive to image noise and to interference among multiple models (Grimson and Huttenlocher, 1990).

In the *indexing methods* invariant representations for both feature sets are computed. In geometric hashing (Lamdan *et al.*, 1990) a randomly chosen noncollinear triplet of points serves as the two-dimensional affine basis for computing the affine invariant coordinates of all the remaining points. These invariant coordinates are the hash-table addresses where the basis triplet's index is stored. Thus for a given affine basis, the invariant representation of the reference set is distributed over several hash-table cells. The procedure is repeated for all the 3-tuples in the reference set. Next, for a transformed set basis the invariant coordinates of the transformed set are computed. The number of times the transformed set coordinates address hash-table cells containing the index of the same reference set basis are counted. If the count is high enough the two bases are declared in correspondence and the underlying affine transformation is computed. Model backprojection then validates the transformation. For noisy images, the random affine bases generate large number of false coincidences (Lamdan and Wolfson, 1991) and at significant noise levels the invariance property of the representations can also be lost (Grimson *et al.*, 1992).

The above discussed object recognition methods require large amount of computations. For example, the worst case complexity of geometric hashing is O$[n^7]$. Most of this effort, however, is wasted on randomly chosen bases which are not in correspondence and thus only increase the probability of a decision error. Non-corresponding configurations of feature subsets should not be taken into account at subsequent steps. Invariant representations these configurations can be obtained with projective invariants and used to filter out the undesired pairs.

Projective invariants were introduced in computer vision by Weiss (1988) and received considerable attention recently (Mundy and Zisserman, 1992). An invariant is an algebraic function in the parameters of a configuration containing a few features. The value of an invariant changes after a projective transformation only by a transformation dependent constant. For absolute scalar invariants, the ones of interest for computer vision, the value of the constant is one. In the first object recognition system using projective invariants the two invariants defined by two coplanar ellipses were employed to index objects from a model library (Forsyth *et al.*, 1991). The use of complete ellipses puts serious limitations on the class of object which can be represented. On the other hand, ellipse fitting to partial data is very sensitive to noise (Porrill, 1990). Several invariants computed for different types of configurations (coplanar lines, two conics, etc.) were used by Wayner (1991) and Rothwell *et al.* (1992b). The topological relations among the features in the model were used to generate lists of invariants representing the same object. Recognition employed hashing techniques (Rothwell *et al.*, 1992b) or graphs (Wayner, 1991).

To obtain the invariant representation of an arbitrary planar shape several methods were proposed. Semi-differential invariants (Van Gool *et al.*, 1991; Brill *et al.*, 1992) are based on low-order derivatives computed at a few points on the boundary curve. The more points are considered (up to four) the less derivatives are needed. Rothwell *et al.* (1992a) used four points on the boundary curve to define a canonical frame in which an invariant representation of a concavity can be obtained. The four points were delineated by tangency, a projectively invariant property. All the mentioned semi-differential invariant representations assume that the correspondence was already found between the points used as anchors.

Invariant representation of a configuration of k features is a mapping from the high-dimensional space spanned by the independent parameters of the configuration to the low-dimensional space of its invariants. For example, two conics in a plane have ten degrees of freedom but yield only two invariants. The mapping is from a ten-dimensional space to a two-dimensional one and different pairs of conics can have very similar invariants. Thus, when an invariant representation is used to select corresponding subsets of features (i.e., to index a data base) the existence of matching errors must be taken into account. These errors were not considered in the invariance based object recognition systems mentioned above.

In this paper we describe a general method for finding in the presence of matching errors between subsets of features the reference/transformed set correspondence. In Section 2 the importance of permutation invariants for object recognition is emphasised and as an example the projective and permutation invariant representation of five coplanar points/lines is introduced. In Section 3 the new feature correspondence algorithm is described and its performance bounds are given. Simulation results are shown in Section 4, and the issues to be faced in applications are discussed in Section 5.

2. Transformation and Permutation Invariants

A formal definition of an absolute scalar projective invariant $\mathrm{I}[\cdot]$ computed with k features in correspondence from the reference set $\mathbf{r}_1, \ldots, \mathbf{r}_k$ and the transformed set $\mathbf{T}(\mathbf{r}_1), \ldots, \mathbf{T}(\mathbf{r}_k)$ is

$$\mathrm{I}[\mathbf{r}_1, \ldots, \mathbf{r}_k] = \mathrm{I}[\mathbf{T}(\mathbf{r}_1), \ldots, \mathbf{T}(\mathbf{r}_k)] \;, \tag{1}$$

where $\mathbf{T}(\cdot)$ is the transformation applied to the reference set. Thus, $\mathrm{I}[\cdot]$ provides an invariant representation for a subset of k features. The invariants currently used in computer vision are feature order sensitive, i.e., their value depend on the order in which the elements $\mathbf{r}_i$ or $\mathbf{T}(\mathbf{r}_i)$ were considered. Matching a given reference and transformed k-tuple pair with an order sensitive invariant representation requires that invariants must be computed for all the $k!$ permutations of one of the k-tuples.

Let $\Pi\{\mathbf{r}_1, \ldots, \mathbf{r}_k\}$ describe a permutation of the k features. An invariant will be called *transformation and permutation* (TP) invariant when it satisfies the condition

$$\mathrm{I}[\mathbf{r}_1, \ldots, \mathbf{r}_k] = \mathrm{I}[\Pi\{\mathbf{r}_1, \ldots, \mathbf{r}_k\}] = \mathrm{I}[\Pi\{\mathbf{T}(\mathbf{r}_1), \ldots, \mathbf{T}(\mathbf{r}_k)\}] \,. \tag{2}$$

A TP-invariant discards not only the influence of the feature transformation $\mathbf{T}$ on the k-tuple but also the influence of the order in which the k features were chosen into that k-tuple. A TP-invariant matching criterion maximizes the probability of matching between k-tuples chosen randomly from the two sets. Let the two sets contain n features in correspondence. Assuming uniform sampling, the probability of matching a given reference k-tuple with a randomly chosen transformed k-tuple is either

$$\mathrm{P}_T = \frac{1}{k!\begin{bmatrix} n \\ k \end{bmatrix}} \quad \text{or} \quad \mathrm{P}_{TP} = \frac{1}{\begin{bmatrix} n \\ k \end{bmatrix}} \,, \tag{3}$$

where P_T is for using an order sensitive invariant representation, and P_{TP} when a TP-invariant representation is employed. The higher probability P_{TP} increases the number of matchings for the same number of trials and thus provides a larger ensemble of matched k-tuples.

Definition of TP-invariants is immediate for the Euclidean group of transformations. The magnitude of an angle, the length of an unoriented segment, do not depend on the order in which the points were chosen and therefore are TP-invariants. For the planar projective group, TP-invariants can be constructed by exploiting the properties of the cross-ratio. In one dimension the cross-ratio of four points A_1, A_2, A_3, A_4 on a line (Fig 1a) is defined as

$$\lambda = (A_1A_2A_3A_4) = \frac{A_1A_3}{A_3A_2} : \frac{A_1A_4}{A_4A_2} \,, \tag{4}$$

where the lengths of the oriented segments A_iA_j are signed. The four points can be considered in $4! = 24$ different orderings which yield six different cross-ratio values (Springer, 1964; p.14):

$$\lambda_1=\lambda \quad \lambda_2=\frac{1}{\lambda} \quad \lambda_3=\frac{\lambda-1}{\lambda} \quad \lambda_4=\frac{\lambda}{\lambda-1} \quad \lambda_5=\frac{1}{1-\lambda} \quad \lambda_6=1-\lambda\ . \tag{5}$$

The six expressions in (5) form a group since any λ_j can be obtained from any λ_i by a sequence of simple transformations.

Proposition 2.1

Any symmetric function of $\lambda_1, \ldots, \lambda_6$ is a TP-invariant of four collinear points under the group of one-dimensional projective transformations.

A function is called symmetric if it is not sensitive to permutations of its variables. For example, the sum of the variables, the sum of all pairwise products of the variables, etc., are symmetric functions. The 24 different point orderings on the line yield the six cross-ratio values in (5). Therefore any symmetric function of $\lambda_1, \ldots, \lambda_6$ will be insensitive to the order in which the points were considered when computing the cross-ratio. Hartshorne (1983, p.317) used the j-invariant

$$J(\lambda) = \frac{(\lambda^2-\lambda+1)^3}{\lambda^2(\lambda-1)^2} \tag{6}$$

when defining elliptic curves. The expression of the j-invariant remains unchanged when λ is substituted with any λ_i from (5). The j-invariant was introduced in computer vision by Maybank (1992) in a study of the invariant properties of noncoplanar conics. However, any symmetric function can be used. For example, the expression $\sum_{i=1}^{6}|\lambda_i|$ is a TP-invariant since a different ordering of the points just changes the order of the terms in the sum. (The symmetric function $\sum_{i=1}^{6}\lambda_i=3$ is a trivial TP-invariant.) Similarly,

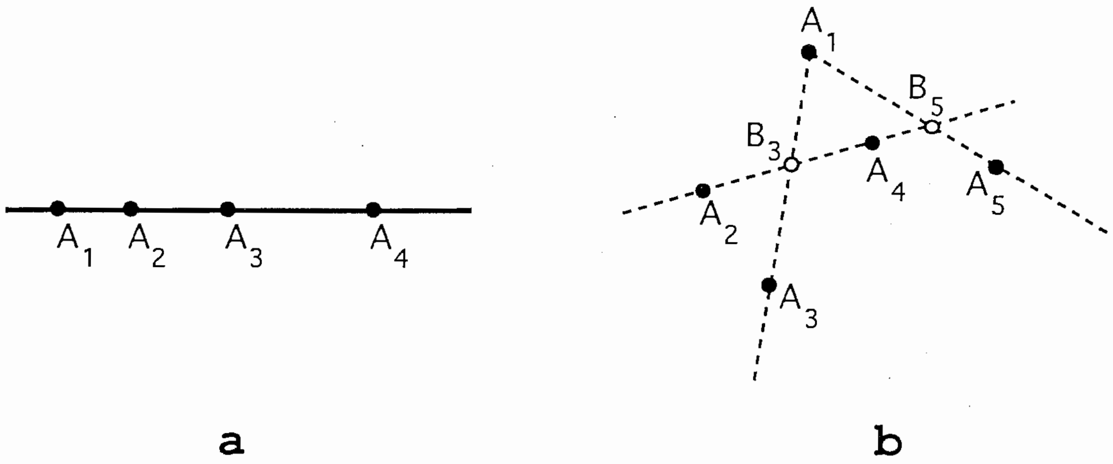

Figure 1
Definition of cross-ratios. a) In one dimension, four collinear points. b) In two dimensions, five points in arbitrary positions in a plane.

$$\sum_{i=1}^{6} \sum_{j=1, j \neq i}^{6} \lambda_i \lambda_j = \frac{\lambda^6 - 3\lambda^5 + 5\lambda^3 - 3\lambda + 1}{\lambda^2 (\lambda - 1)^2} \tag{7}$$

is also a TP-invariant. Beside computational considerations none of the TP-invariants defined above seems to be more advantageous.

The importance of cross-ratio for computer vision was already recognized by Duda and Hart (1973, pp. 407-414). The projective transformation of a plane is defined by the correspondence of four points in arbitrary positions (no three collinear). Therefore, to generalize the cross-ratio in two dimensions a five-point configuration is necessary. Several equivalent definitions can be given (see for example Barrett *et al.*, 1991) but for us the procedure shown in Figure 1b is the most convenient. One of the five points (A_1) is chosen as the center of perspectivity and a line passing through two other points (A_2 and A_4) as a one-dimensional projective space. The one-dimensional cross-ratio ($A_2 B_3 A_4 B_5$) is then taken as the two-dimensional cross-ratio associated with A_1. The coordinates of the intersection points

B_3 and B_5 are not needed explicitly, the two-dimensional cross-ratio can also be computed using ratios of areas of triangles defined by the points A_i. The procedure is repeated for all the points A_2 to A_5 as center of perspectivity. Of the obtained five cross-ratio values only two are independent, and they are known as the projective coordinates of (say) A_1. Like in the one-dimensional case the two-dimensional cross-ratio is not order invariant. Changing the order of the points in Figure 1b yields different two-dimensional cross-ratio values, and so does choosing a different line as the one-dimensional projective space. Haralick (1989) gave the relations among the two-dimensional cross-ratios obtained with different orderings of the five points.

Proposition 2.1 enables construction of a TP-invariant representation for 5-tuples of points or lines. (Points and lines are dual in projective spaces.) First, the five two-dimensional cross-ratios are computed for the five points. Any of the symmetric functions mentioned before can be used as TP-invariant and each point is associated with the value of that function for the its cross-ratio. The TP-invariant is insensitive to the order of the four points (in our example, A_2, B_3, A_4, B_5). This insensitivity removes the influence of the way the configuration was labeled or which line was chosen to compute a cross-ratio. Since the cross-ratios are not independents, the five TP-invariant values associated with the five points provide a redundant representation of the configuration. We are currently investigating the properties of this representation and its usefulness for the recognition of planar objects.

3. Feature Correspondence Algorithm

The algorithm is described using an idealized feature selection model. This simple model allows us to find the necessary conditions for reliable performance and to emphasize the issues that must be faced in more realistic conditions. Let assume for the moment that both the reference and the transformed set contains n features in correspondence. A TP-invariant matching criterion is used to select the k-tuples whose features are in correspondence. All the n features have equal probability to be in any of the k positions of a k-tuple satisfying the matching criterion. A matching error appears whenever a feature in the reference k-tuple does not have its correspondent in the associated transformed k-tuple. Matching errors are modeled by allowing u features in the k-tuples drawn from the transformed set to be incorrect with probability α. Thus, $u \leq k < n$ and $n \geq k+u$. Given the feature $\mathbf{r}_i$ in the reference k-tuple, the probability that its correspondent $\mathbf{T}(\mathbf{r}_i)$ is in the matched transformed k-tuple is

$$\gamma_{ii} = 1 - \frac{u}{k}\alpha \ . \tag{8}$$

The probability that a second feature $\mathbf{T}(\mathbf{r}_j)$ was also chosen into the same transformed k-tuple can be computed considering the two possibilities:

- The feature $\mathbf{r}_j$ is in the reference k-tuple and therefore the feature $\mathbf{T}(\mathbf{r}_j)$ should be in the transformed k-tuple

  $$\theta_1 = \frac{k-1}{n-1}\left[1 - \frac{u}{k}\alpha\right] \ , \tag{9}$$

  where the first factor is the probability that $\mathbf{r}_j$ is selected into the reference k-tuple given that $\mathbf{r}_i$ was already chosen into.

- The feature $\mathbf{r}_j$ is not in the reference k-tuple, that is, the feature $\mathbf{T}(\mathbf{r}_j)$ is in the transformed k-tuple due to an error

  $$\theta_2 = \left[1 - \frac{k-1}{n-1}\right]\frac{u\alpha}{n-k} \ . \tag{10}$$

Thus, the probability that given the presence of a feature $\mathbf{r}_i$ in the reference k-tuple, both $\mathbf{T}(\mathbf{r}_i)$ and $\mathbf{T}(\mathbf{r}_j)$ are present in the matched transformed k-tuple is

$$\gamma_{ij} = \theta_1 + \theta_2 = \frac{k-1}{n-1}\left[1 - \frac{u}{k}\alpha\right] + \frac{u\alpha}{n-1} \ . \tag{11}$$

We can describe now the new feature correspondence algorithm. The tool to extract the individual feature correspondence from the ensemble of matched k-tuples is a contingency table. The labels of the reference set features are the row addresses in the table and the labels of the transformed set features are the column addresses. Thus any reference and transformed feature pair uniquely defines a cell in the contingency table.

Feature Correspondence Algorithm

```
1. Arbitrarily label the features in the two sets.
2. Obtain an ensemble of matched $k$-tuples satisfying the TP-invariance criterion.
3. For every matched $k$-tuple:
    For every feature in the reference $k$-tuple:
        3.1. Cast a vote on every feature in the transformed set $k$-tuple.
        3.2. Record the votes in the contingency table.
4. Extract the feature correspondence from the contingency table:
    4.1. The cell with the highest number of votes gives the labels of the feature pair
         with the most often recorded correspondence.
    4.2. Remove from the table the column and the row of the cell found at 4.1.
    4.3. If the table is not empty return to 4.1.
```

When the reference feature $\mathbf{r}_i$ votes on the transformed features in the matched k-tuple, the probability that he votes for its correspondent $\mathrm{T}(\mathbf{r}_i)$ is γ_{ii}. Any one of the $n-1$ non-corresponding features $\mathrm{T}(\mathbf{r}_j)$ receives a vote with probability γ_{ij}. If in t trials (matchings) the feature $\mathbf{r}_i$ is chosen q times, the expected value in the cell $(\mathbf{r}_i, \mathrm{T}(\mathbf{r}_i))$ is the closest integer to $q\gamma_{ii}$. Any cell $(\mathbf{r}_i, \mathrm{T}(\mathbf{r}_j))$, $j \neq i$, has the expected value the closest integer to $q\gamma_{ij}$. However, these are only expected values and their fluctuation can come from several sources. The number of times a reference feature is chosen into a k-tuple in a given number of trials has a binomial distribution with parameter $\mu = k/n$. Thus $q = [t\mu]$ where $[\cdot]$ is the integer part of the argument, and has the variance $t\mu(1-\mu)$. Due to the matching errors, the number of correct correspondences in the reference/transformed k-tuple pair is also a binomial random variable with parameter α. Note that when the number of trials is small the effect of sampling nonuniformity yielding deviations from the binomial distributions should also be taken into account.

The correspondence of features $\mathbf{r}_i$ and $\mathrm{T}(\mathbf{r}_i)$ can be extracted from the contingency table if the number of votes in cell $(\mathbf{r}_i, \mathrm{T}(\mathbf{r}_i))$ exceeds the largest number of votes in any cell $(\mathbf{r}_i, \mathrm{T}(\mathbf{r}_j))$, $j \neq i$. The number of votes are random variables and we must have

$$\mathrm{Prob}\,[\,\min \Gamma_{ii} > \max_{j \neq i} \Gamma_{ij}\,] = 1 \qquad i = 1, \ldots, n \ . \tag{12}$$

where the capital letters signify the randomness of γ_{ii} and γ_{ij}. A practical (but approximative) way to reformulate (12) is

$$\gamma_{ii} > C\gamma_{ij} \ , \tag{13}$$

where the constant $C > 1$ accounts for the spread of the central value. For example, assuming a one-sided range of 10 percent from the mean for both γ-s yields $C = 1.22$. Let characterize the quality of the matchings (the presence of matching errors) with

$$\beta = \frac{u}{k}\alpha \tag{14}$$

and condition (13) becomes

$$1-\beta > C\left[\frac{k-1}{n-1}(1-\beta)+\frac{k\beta}{n-1}\right] \tag{15}$$

or

$$n > \frac{kC}{1-\beta} + 1 - C\ . \tag{16}$$

Given the number of features used in the TP-invariant matching criterion and the quality of the matchings, the inequality (16) shows the smallest reference/transformed set size required for reliable performance. In Figure 2 the graphs of this bound are shown function of β for three values of $k=2,4,6$. The graphs are the prototype curves for $C=1$. The parameter C, however, is only a scaling factor since the second term at the right side of (16) has reduced

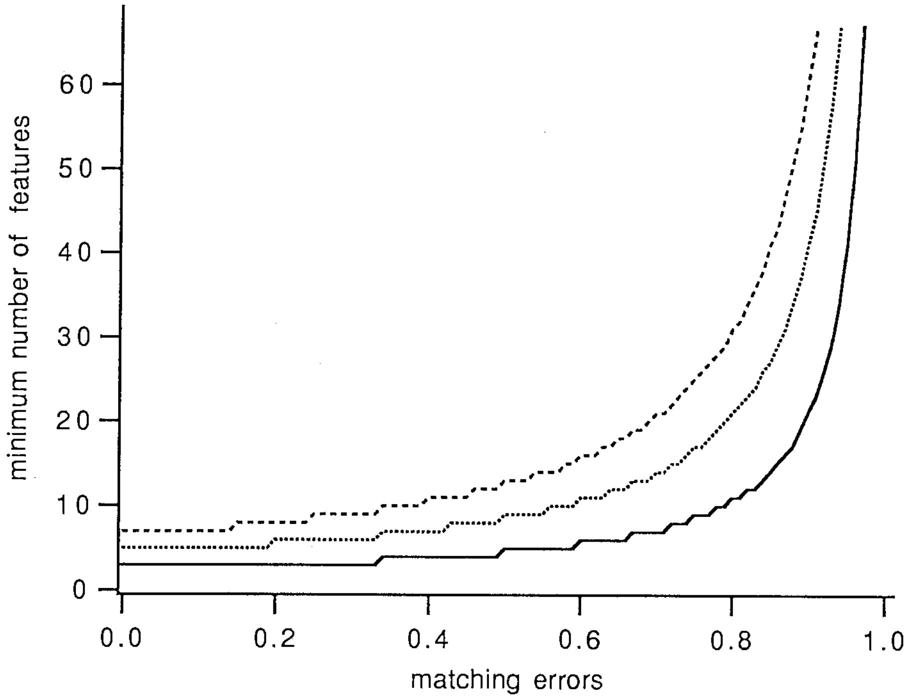

Figure 2

Minimum number of features necessary for correspondence recovery function of the k-tuple matching errors. Solid line: $k=2$. Dotted line: $k=4$. Hashed line: $k=6$.

influence.

The inequality (16) for reliable performance can be interpreted in several ways. Robustness of the correspondence algorithm is measured by the size of the $\gamma_{ii} - C\gamma_{ij}$ difference. The larger this difference the less the probability of not satisfying (12). Given the data (i.e., n and β) the difference increases when the matching criterion uses a smaller k. Since the value of k is bounded downward by the transformation group under which TP-invariance must be assured, the following result was obtained:

Proposition 3.1

> The TP-invariant matching criterion should be based on the least number of features allowed by the given transformation group.

Pairs of points can be enough for seeking correspondence under Euclidean transformations, but at least five points are required when the correspondence is sought under the planar projective group.

The quality of the matchings is represented only through β and therefore we have:

Proposition 3.2

> The same performance is obtained when in a transformed k-tuple all the features are erroneous with a low probability, or only a few of them are incorrect with a high probability.

In the frequently met case of occlusions the transformed set contains only a partial instance of the reference set. Reliable extraction of feature correspondence in the presence of occlusions requires low noise level in the transformed set. (Recall the reference set was assumed to be always correct.) Indeed, some of the features without correspondent pass the matching criterion, yielding a small u and relative large α. When the features in the transformed set are also corrupted by significant noise more matching errors occur. Increase in both u and α decreases the difference $\gamma_{ii} - C\gamma_{ij}$ and therefore deteriorates the performance.

In some applications the number of features in the reference and transformed sets can be increased. For example, the available features can be used as anchors to delineate other features in their neighborhood. Such procedure is recommended.

Proposition 3.3

> The performance of the feature correspondence algorithm improves when the number of features whose correspondence is sought increases.

Indeed, given k and β, if more corresponding features are available (without knowning their correspondence of course) the inequality (16) can always be satisfied. At least theoretically, the solution of the correspondence problem can be found for arbitrarily high levels of matching errors. Note that Proposition 3.3 is not related to the model backprojection procedure often used in the object recognition methods to validate hypotheses.

It is important to emphasize that Propositions 3.1 to 3.3 use the uniform feature selection hypothesis assumed throughout this section. That is, any change in the data (like increasing the number of features) should not introduce significant deviations from the assumed uniform sampling or decrease the quality of matchings. In Section 5 we discuss the issues raised when the algorithm is to be employed in real applications.

4. Simulation Results

The influence of random sampling on the performance of the algorithm was investigated through simulations. The reference and the transformed sets contained ten features ($n = 10$)

and correspondence was defined by two features having the same label. The TP-invariant criterion was assumed to be based on 5-tuples of features ($k=5$), the minimum required for the group of planar projective transformations. The matching errors were distributed over the entire 5-tuple ($u=5$) with probability α and thus $\beta=\alpha$.

The following procedure was repeated t times. First the labels of a reference 5-tuple were drawn at random from the set $\{1, \ldots, 10\}$. Each of the five elements were then changed with probability α into an element not in the reference 5-tuple. The obtained five labels defined the matched transformed 5-tuple. The values of all the contingency table cells addressed by the pair of reference and transformed k-tuples were increased by one.

The distribution of γ_{ii} was measured by computing the mean and the standard deviation of the values in the cells (i,i), $i=1, \ldots, 10$. The results were expressed as a probability by normalizing the computed parameters with q, the expected number of selection of label i into a k-tuple. There were no significant differences among the γ_{ij} distributions for different i-s and the cells $(1,j)$, $j=2, \ldots, 10$ were used to measure the mean and the standard deviation. In a transformed k-tuple u features can be incorrect with probability α, and therefore the distribution of the number of correct feature correspondences in a matched k-tuple pair is

$$\text{Prob}\,[\,(k-x)\text{ out of }k\text{ correct}\,] = \begin{cases} \begin{bmatrix} u \\ x \end{bmatrix} \alpha^{x}\,(1-\alpha)^{u-x} & x=0,1,\ldots,u \\ 1 & x=u+1,\ldots,k\,. \end{cases} \tag{17}$$

This distribution was also measured during the experiments.

In Figure 3 the results obtained for 100 trials with $\alpha=0.3$ are shown. Almost all the matched k-tuples contain at least one error as can be seen from the distribution of the number of correct correspondences. The number of trials is too small and the effect of nonuniform sampling yields large standard deviations for γ_{ii} and γ_{ij}. Assuming the spread from the

MATCHING STATISTICS

Probability of the number of correct correspondences.

	0	1	2	3	4	5
Theory	0.002	0.028	0.132	0.309	0.360	0.168
Simulation	0.010	0.030	0.060	0.320	0.360	0.220

CONTINGENCY TABLE

Rows: reference feature labels. Colums: transformed feature labels.

No.	1	2	3	4	5	6	7	8	9	10
1	31	23	17	26	20	19	27	25	24	23
2	19	35	23	18	21	26	25	27	25	21
3	19	22	34	18	25	23	25	21	28	25
4	14	19	19	30	16	20	20	21	21	20
5	23	17	24	15	34	20	30	23	24	25
6	24	32	26	23	26	40	27	30	23	29
7	24	19	24	18	23	25	35	26	28	28
8	21	25	23	18	24	28	23	39	28	26
9	22	25	25	21	26	26	28	29	41	32
10	18	23	25	28	30	23	30	29	38	46

PERFORMANCE ROBUSTNESS

Normalized number of votes.

	Theory	Simulation
Correct: γ_{ii}	0.7000	0.7228 ± 0.0976
Incorrect: γ_{ij}	0.4778	0.4697 ± 0.0657

Figure 3

Simulation example. Parameters: $n=10$, $k=5$, $u=5$, $\beta=\alpha=0.3$, $t=100$. Identical labels are a correct correspondence.

central value as being twice to three times the standard deviation, there is an overlap between the two distributions. The condition (12) may be violated but the correct correspondence is still recovered as can be seen from the contingency table.

The largest value in the table is 46 in cell $(10, 10)$. The tenth row and column are thus removed from the table and the next largest value, 41, is found in cell $(9, 9)$. The ninth row and column are also removed, and the cell $(6, 6)$ holds now the currently largest value, 40, etc. The *marker sequence* of largest values identifying the corresponding elements is:

$$46, 41, 40, 39, 35, 35, 34, 34, 31, 30.$$

Two important observations should be made. A necessary condition for correspondence recovery is that all the cells in the marker sequence have different values. If the currently largest value in the contingency table is appears in more than one cell it is impossible to define a unique correspondence. However, ties in the marker sequence can always be broken with additional trials.

The marker sequence guides a peeling-off procedure of the contingency table when the feature correspondence is extracted. The technique is based on the observation that if a correspondence is correctly identified none of the two features can be paired with any other. Thus, the analysis of the contingency table obeys the Optimality Principle in dynamic programming: at every stage of the analysis the decision is optimal given the decisions at the previous steps. When the sampling nonuniformity is significant (number of trials too small) some cells in the table can have values which are larger than those in the marker sequence. In our example the cell $(10, 9)$ has 38, $(9, 10)$ has 32, and four cells have the value 30 beside $(4, 4)$. The large values appear since a more frequently chosen reference feature will also cast more incorrect votes on a given transformed feature. When the peeling-off procedure

MATCHING STATISTICS

Probability of the number of correct correspondences.

	0	1	2	3	4	5
Theory	0.031	0.156	0.312	0.312	0.156	0.031
Simulation	0.029	0.161	0.294	0.313	0.167	0.036

RECOVERED CORRESPONDENCE

Identical labels are a correct correspondence.

Feature Label		Cell
Reference Set	Transformed Set	Value
1	7	285
7	9	276
5	2	275
6	8	270
8	6	267
9	3	264
3	5	254
10	10	247
4	4	236
2	1	228

PERFORMANCE ROBUSTNESS

Normalized number of votes.

	Theory	Simulation
Correct: γ_{ii}	0.5	0.5067 ± 0.0213
Incorrect: γ_{ij}	0.5	0.4987 ± 0.0236

Figure 4

Simulation example. Parameters: $n=10$, $k=5$, $u=5$, $\beta=\alpha=0.5$, $t=1000$.

removes the selected row and column from the table the cells having large values due to this artifact are also eliminated. Note that condition (12) is to be satisfied for each i separately.

Assuming a ten percent spread ($C = 1.22$), the smallest number of features for which correspondence can still be recovered is $n = 9$. For $n = 10$ the largest tolerated β is 0.4. The example in Figure 3 illustrates the problems created by a small number of trials. While the algorithm always recovers the correspondence, its performance cannot be called robust since a small increase of β can cause its breakdown. To obtain a robust performance it suffices to increase the number of trials. Using 1000 matchings yields $\gamma_{ii} = 0.7063 \pm 0.0315$ and $\gamma_{ij} = 0.4765 \pm 0.0253$. The overlap between the two distributions is now negligible and condition (12) is always satisfied. The ten largest number of votes in the contingency table are in the ten cells indicating the corresponding features. The probability of having a tie is also very low. Any further increase in the number of trials is unnecessary.

In Figure 4 the simulation results for $\beta = \alpha = 0.5$ and 1000 trials are shown. The matching errors exceed the upper bound for β and most of the matched k-tuple pairs contain only two or three correct correspondences. The distributions of γ_{ii} and γ_{ij} are identical. Increasing the number of trials will only reduce their spread but cannot fulfill condition (12). The contingency table does not reflect the correspondence between the reference and transformed sets. Once an incorrect correspondence is extracted the error will propagate since a cell which indicated a correct correspondence was also removed. All the correct correspondences shown in Figure 4 were found by chance and not by a systematic procedure. For $k = 5$, $\beta = 0.5$ and $C = 1.22$ the smallest number of features required for reliable performance is obtained from (16) as $n = 13$. When 13 features are used in the algorithm $\gamma_{ii} = 0.5073 \pm 0.0350$ and $\gamma_{ij} = 0.3740 \pm 0.0246$. The decrease of γ_{ij} allows the algorithm to always recover the correct correspondences. Note that in spite of using 13 instead of 10 corresponding features the number of trials and thus the amount of computation remained the same.

5. Discussion

The new correspondence algorithm has modular structure and a well defined condition (12) for reliable performance. Once the condition cannot be satisfied the performance deteriorates steeply. The algorithm belongs to the class of engineering techniques whose adoption in computer vision was strongly recommended by Haralick (1992). The robustness of the algorithm is achieved through combining many outcomes of the same simple process (matching of k-tuples) to extract information (feature correspondence) buried under the noise. We have shown elsewhere that such a processing paradigm is useful in computer vision where assumptions about underlying models often cannot adequately represent the data (e.g., Mintz *et al.*, 1992). The new correspondence algorithm does not have embedded any assumption about distributions unlike the optimized geometric hashing method of Costa *et al.* (1990). The performance bounds in Section 3 were computed for a simple feature selection model which is unrealistic in applications. However, as long as (12) is true, feature correspondence will be recovered independent of the distribution of the matching errors.

Many of the ideas presented in this paper were crystalized while trying to use the two projective invariants defined by two planar conics for feature correspondence recovery (Meer and Weiss, 1992). Five points (or lines) uniquely define a conic in a plane. Two conics have two invariants under the planar projective transformation group. These two invariants were used as the matching criterion between 6-tuples of points randomly chosen from the reference and transformed sets. The six points were divided into a subset of four and a subset of two. The two conics were defined by taking the four points and a point from the second subset. Note that the order of points in either subset is not important and we had a partially TP-invariant matching criterion.

To recover the correspondence often an unrealistic feature accuracy was required: point coordinates in the range of one hundred with at least three correct decimal values. Since the five points defining a conic were in arbitrary positions, the conic matrices (used in computing the invariants) could be ill-conditioned. Then the smallest change in the position of one of the points can yield a significant change in the parameters of the conic. The two conics have ten degrees of freedom but the two invariants span only a two-dimensional space. This projection into a lower dimensional space is the main source of the matching errors. Non-corresponding pairs of conics can have very similar invariant values and thus many 6-tuples were matched with a high probability of error. Increasing the number of corresponding points did not help since the probability of having quasi-degenerate conics and thus the probability of erroneous matchings increased too. Note that from the performance analysis in Section 3 matching should be based on 5-tuples of features and not on two conics.

In Section 4 it was shown that the performance of the algorithm can be improved by increasing the number of features in correspondence. The probability of the matching errors, however, should not increase significantly with the number of features. If the number of matchings remains the same, the expected number of selections of feature $\mathbf{r}_i$ into a k-tuple decreases. In t trials a feature is chosen on the average $q = \left\lceil \frac{t\,k}{n} \right\rceil$ times and an increase in the number of matchings can compensate for the increase of n. Matching of k-tuples requires constant time and the number of trials is (weakly) dependent on n. The analysis of the contingency table is $\mathrm{O}[n^2]$. We conclude that the complexity of the feature correspondence algorithm is $\mathrm{O}[n^2]$. The low complexity of the new feature correspondence algorithm is similar to that of other invariant indexing function methods (e.g., Rothwell *et al.*, 1992b). These techniques do not process all the features in the reference or the transformed set together when looking for a model instance, as it is done in geometric hashing. Therefore invariance based indexing is better suited for multiple-model object recognition tasks.

While retaining the good properties of the invariant indexing function methods, the principle behind the new correspondence algorithm is different. In the former a small number of adjacent configurations (topology is preserved under projective transformation) are used to build the invariant representation of an object. In the correspondence algorithm the features (which can also be grouped into adjacent regions) are matched in many configurations to create the statistically significant support for the correspondences. The possibility of erroneous matchings is embedded in the algorithm. Validation through backprojection of an extended model (which is a very expensive procedure) can be eliminated. Let the correspondence of n planar features returned by the algorithm. There are $\binom{n}{4}$ possibilities to compute the transformation between the reference and transformed sets. The correspondence is correct when several combinations give similar transformations. Recall that the performance of the algorithm deteriorates very steeply once the condition (12) is not satisfied.

Application of the correspondence algorithm in a real situation should be preceded by the analysis of the matching errors generated in the task. The performance is limited by the dependence of these errors on the amount of noise in the data and/or on the number of non-corresponding features introduced by occlusions. Inaccurate data will always yield matching errors since an invariant representation of k features is a many-to-one mapping. The usefulness of invariants in building fast and robust object recognition systems is contingent upon understanding their sensitivity to noise.

6. Conclusion

A new approach toward solving the feature correspondence problem was presented. Small subsets of features are matched through similar invariant representations. The probability of finding a match is maximized since invariant representations which also tolerate permutations of the elements are used. The existence of matching errors is explicitly taken into account and their effect is eliminated by pooling the information from the entire ensemble of matched subsets into a contingency table. The condition for robust performance is well defined and it should be checked before employing the algorithm in applications.

Acknowledgment

The opportunity to spend three wonderful months at the Auditory and Visual Perception Laboratories of the Advanced Telecommunications Research Institute International, Kyoto, Japan, and the support of K. Shimohara, N. Sonehara and E. Yodogawa are gratefully acknowledged.

References

[1] E.B. Barrett, P.M. Payton, N.N. Haag, M.H. Brill (1991): General methods for determining projective invariants in imagery. *CVGIP: Image Understanding,* **53,** 46–65.

[2] M.H. Brill, E.B. Barrett, P.M. Payton (1992): Projective invariants for curves in two and three dimensions. In [18], 193–214.

[3] M.S. Costa, R.M. Haralick, L.G. Shapiro (1990): Optimal affine-invariant point matching. In *Proceedings of the 10th International Conference on Pattern Recognition,* Atlantic City, NJ, June 1990, 233–236.

[4] R.O. Duda, P.E. Hart (1973): *Pattern Classification and Scene Analysis.* Wiley, New York.

[5] D. Forsyth, J.L. Mundy, A. Zisserman, C. Coelho, A. Heller, C. Rothwell (1991): Invariant descriptors for 3-D object recognition and pose. *IEEE Trans. Pattern Anal. Mach. Intell.* **PAMI-13,** 971–991.

[6] W.E.L. Grimson, D.P. Huttenlocher (1990): On the sensitivity of Hough transform for object recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **PAMI-12,** 255–274.

[7] W.E.L. Grimson, D.P. Huttenlocher, D.W. Jacobs (1992): A study of affine matching with bounded sensor error. In *Computer Vision–ECCV '92,* G. Sandini (Ed.), Springer-Verlag, Berlin, 292–306.

[8] R.M. Haralick (1989): Monocular vision using the cross-ratio. In *Proceedings of The 6th Scandinavian Conference on Image Analysis,* Oulu, Finland, June 1989, 943–952.

[9] R.M. Haralick (1992): Performance characterization in image analysis: thinning a case in point. *Patter Recognition Letters,* **13,** 5–12.

[10] R. Hartshorne (1983): *Algebraic Geometry.* Corrected third printing. Springer-Verlag, New York.

[11] D.P. Huttenlocher, S. Ullman (1990): Recognizing solid objects by alignment with an image. *International Journal of Computer Vision,* **5,** 195–212.

[12] Y. Lamdan, J.T. Schwartz, H.J. Wolfson (1990): Affine invariant model-based object recognition. *IEEE Trans. Robot. Autom.* **6,** 578–589.

[13] Y. Lamdan, H.J. Wolfson (1991): On the error analysis of 'Geometric hashing'. *Proceedings of the 1991 IEEE Computer Society Conference on Computer Tision and Pattern Recognition,* Lahaina, Maui, Hawaii, June 1991, 22–27.

[14] S.J. Maybank (1992): The projection of two non-coplanar conics. In [18], 105–119.

[15] P. Meer, I. Weiss (1992): Point/line correspondence under 2D projective transformation. To appear in *Proceedings of the 11th International Conference on Pattern Recognition: Computer Vision and Applications,* The Hague, The Netherlands, September 1992.

[16] D. Mintz, P. Meer, A. Rosenfeld (1992): Analysis of the least median of squares estimator for computer vision applications. In *Proceedings of the 1992 IEEE Computer Society Conference on Computer Vision and Pattern Recognition,* Champaign, Illinois, June 1992, 621–624.

[17] J.L. Mundy and A.J. Heller (1990): The evolution and testing of a model-based object recognition system. *Proceedings of the Third International Conference on Computer Vision,* Osaka, Japan, December 1990, 268–282.

[18] J.L. Mundy, A. Zisserman (Eds.) (1992): *Geometric Invariance in Machine Vision.* MIT Press, Cambridge, Mass, 1992.

[19] J. Porrill (1990): Fittind ellipses and predicting confidence envelopes using a bias corrected Kalman filter. *Image and Vision Computing* **8,** 37–41.

[20] C.A. Rothwell, A. Zisserman, D.A. Forsyth, J.L. Mundy (1992a): Canonical frames for planar object recognition. In *Computer Vision–ECCV '92,* G. Sandini (Ed.), Springer-Verlag, Berlin, 757–772.

[21] C.A. Rothwell, A. Zisserman, J.L. Mundy, D.A. Forsyth (1992b): Efficient model library access by projective invariant indexing functions. In *Proceedings of the 1992 IEEE Computer Society Conference on Computer Vision and Pattern Recognition,* Champaign, Illinois, June 1992, 109–114.

[22] C.E. Springer (1964): *Geometry and Analysis of Projective Spaces.* W.H. Freeman and Company, San Francisco.

[23] L. Van Gool, O. Kempenaers, P. Oosterlinck (1991): Recognition and semi-differential invariants. In *Proceedings of the 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition,* Maui, Hawaii, June 1991, 454–460.

[24] P.C. Wayner (1991): Efficiently using invariant theory for model-based matching. In *Proceedings of the 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition,* Lahaina, Maui, Hawaii, June 1991, 473–478.

[25] I. Weiss (1988): Projective invariants of shapes. Computer Vision Laboratory. University of Maryland, College Park, CAR-TR-339, January 1988.