

A Distance Metric for Multidimensional Histograms

MICHAEL WERMAN,* SHMUEL PELEG,* AND AZRIEL ROSENFELD†

Center for Automation Research, University of Maryland, College Park, Maryland 20742

Received November 28, 1984; revised April 2, 1985; accepted June 12, 1985

A metric is defined on the space of multidimensional histograms. Such histograms store in the x th location the number of events with feature vector x ; examples are gray level histograms and co-occurrence matrices of digital images. Given two multidimensional histograms, each is "unfolded" and a minimum distance pairing is performed using a distance metric on the feature vectors x . The sum of the distances in the minimal pairing is used as the "match distance" between the histograms. This distance is shown to be a metric, and in the one-dimensional case is equal to the absolute difference of the two cumulative distribution functions. Among other applications, it facilitates direct computation of the distance between co-occurrence matrices or between point patterns. The problem of finding a translation to minimize the distance between point patterns is also discussed. © 1985 Academic Press, Inc.

1. INTRODUCTION

Multidimensional histograms are often used to characterize images. Co-occurrence matrices [1, 2], for example, are used to characterize textures. In these two-dimensional histograms, given a displacement α the (i, j) entry in the matrix indicates how many pixel pairs at distance α apart have gray levels i and j , respectively. A gray level histogram is a one-dimensional example, where the i th entry indicates the number of pixels having gray level i . An image itself can be regarded as a two-dimensional histogram, if we regard it as a distribution of photons, and bright points indicate high concentration of photons.

A simple way to measure the distance between two multidimensional histograms would be to use their vector distance, defined, e.g., as the sum of the absolute differences between corresponding components. However, this distance measure does not represent the underlying semantics of their features. As an example, consider the following three eight-valued histograms: $H_0 = (16, 0, 0, 0, 0, 0, 0, 0)$ (all 16 pixels are 0); $H_1 = (0, 16, 0, 0, 0, 0, 0, 0)$ (all 16 pixels are 1); and $H_6 = (0, 0, 0, 0, 0, 0, 16, 0)$ (all 16 pixels are 6). Using vector distance, all three histograms are equally far apart from each other. However, the gray level in the image of H_0 is much closer to that of H_1 than to that of H_6 . The vector distance fails to indicate this.

A distance metric for one-dimensional histograms that incorporates similarity between features was suggested by Shen and Wong [3]. This measure is also a metric. It involves "unfolding" the histogram, where each gray level is repeated as many times as the value in its entry in the histogram. The unfolding of $(3, 0, 0, 2, 1)$, for example, is $(0, 0, 0, 3, 3, 4)$. Note that for images of the same size, the sum of the histogram entries must equal the number of pixels in the image; thus all unfolded

*Permanent Address: Dept. of Computer Science, The Hebrew University of Jerusalem, 91904 Jerusalem, Israel. These authors were supported in Israel by a grant from the Israel Academy of Sciences.

†The support of the National Science Foundation under grant DCR-82-18408 is gratefully acknowledged, as is the help of Janet Salzman in preparing this paper.

histograms have the same length. The distance between two histograms is defined as the sum of the pairwise distances between all pairs of points having the same location in the two unfolded histograms. For example, to compute the distance between $A = (3, 0, 0, 2, 1)$ and $B = (2, 1, 0, 0, 3)$ we unfold them into $UF(A) = (0, 0, 0, 3, 3, 4)$ and $UF(B) = (0, 0, 1, 4, 4, 4)$. The distance will then be $0 + 0 + 1 + 1 + 1 + 0 = 3$. We call the above distance “match-distance” as it is the minimal distance of any matching of points in one histogram to points in the second histogram.

In the one-dimensional case the match-distance is identical to the vector distance between the two cumulative distribution functions. In the above example, let \mathbf{A} and \mathbf{B} be the cumulative distributions of A and B , respectively. Thus $\mathbf{A} = (3, 3, 3, 5, 6)$ and $\mathbf{B} = (2, 3, 3, 3, 6)$. Their absolute vector distance is $1 + 0 + 0 + 2 + 0 = 3$, the same as the “unfolded” match-distance.

In general, when the feature space is a subset of the real numbers, and the distance between two feature values is their absolute difference, we can have another equivalent definition for the unfolded match-distance using cumulative distribution functions.

THEOREM 1. *Let $f(x), g(x)$ be two histograms from the real numbers, where a delta function designates the location of a point. (A delta function is a function δ such that $\int_{-\epsilon}^{\epsilon} \delta(x) dx = 1$ for all $\epsilon > 0$.) The requirement of equal numbers of points is represented by $\int_{-\infty}^{\infty} f(x) dx = \int_{-\infty}^{\infty} g(x) dx$. With distances between feature values measured by the absolute difference, the match-distance between f and g is*

$$\int_{-\infty}^{\infty} \left| \int_{-\infty}^x f(t) dt - \int_{-\infty}^x g(t) dt \right| dx.$$

Define $F(x) = \int_{-\infty}^x f(t) dt$, and $G(x) = \int_{-\infty}^x g(t) dt$. $F(x)$ is the number of points in the histogram f having values less than or equal to x , and similarly for $G(x)$. Since in the linear case points are paired sequentially for minimal matching [3], at a point t all possible points with value less than t are paired. $|F(t) - G(t)|$ points, however, will be paired with points having higher value than t . Thus for every t the number of points with value less than t paired with points having values higher than t is $|F(t) - G(t)|$. Therefore $\int_{-\infty}^{\infty} |F(t) - G(t)|$ is equal to the sum of all pairwise distances in a minimal matching.

EXAMPLE. Let $f = (0, 1, 1, 1, 0)$ and $g = (2, 0, 0, 0, 1)$ be two histograms from the set $\{1, 2, 3, 4, 5\}$ to the natural numbers. Their match-distance using unfolding is as follows:

$$UF(f) = \{2, 3, 4\}; UF(g) = \{1, 1, 5\}, \text{ and}$$

$$\rho(f, g) = |1 - 2| + |1 - 3| + |5 - 4| = 4.$$

Using the new definition we have F and G displayed in Fig. 1, with a delta function where a point is located. Their match distance is the shaded area in Fig 1e.

Comparison of histograms by their cumulative distribution functions has a long history [15]. Unfortunately, there is no multidimensional extension to the distance definition using cumulative distribution functions. The matching approach, yielding identical results in one dimension, is extendible to the multidimensional case.

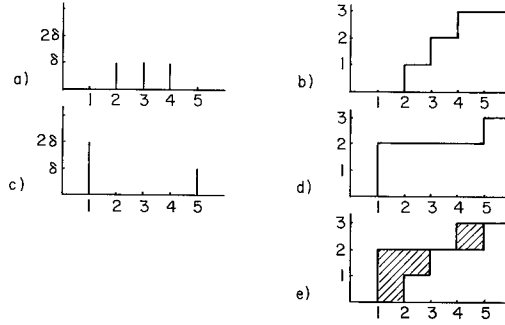


FIG. 1. Match distance as an integral: (a) f , (b) F , (c) g , (d) G , (e) F and G superimposed. Shaded area is the match distance, area = 4.

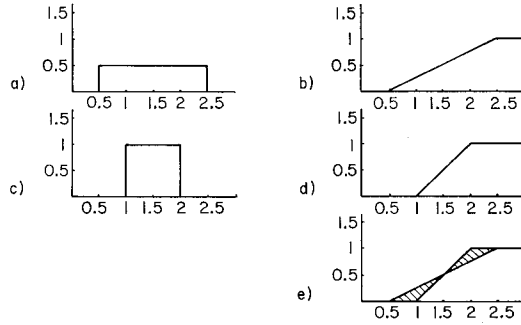


FIG. 2. Distance between distribution functions: (a) the density function $U[0.5, 2.5]$, (b) its distribution function, (c) the density function $U[1, 2]$, (d) its distribution function, (e) the shaded area is the distance between the two distribution functions, area = 0.25.

The above distance definition is immediately extendible to functions $p, q: R \rightarrow R$ such that $\int_{-\infty}^{\infty} p(x) dx = \int_{-\infty}^{\infty} q(x) dx < \infty$. As a special case, this defines a distance metric between distribution functions. Figure 2 shows the distance between two different uniform distributions.

In the following sections we generalize this one-dimensional histogram metric, and introduce a metric between histograms of any dimension, which is also applicable to point patterns. Use of this distance to compare texture features [7] gave better results than vector distance.

2. A METRIC FOR MULTIDIMENSIONAL HISTOGRAMS

Let L be a set of descriptors having the metric μ . Let $f: L \rightarrow N$ be a function from L to the natural numbers, denoting the frequency of occurrence of each descriptor $l \in L$, and satisfying

$$\sum_{l \in L} f(l) = M$$

where M is a constant. For gray level histograms, this restriction implies that only

pictures of the same size M can be compared, and similarly for co-occurrence matrices (ignoring border effects).

DEFINITION 1. $UF(f)$, the unfolding of f , is the multiset of elements of L , each $l \in L$ appearing $f(l)$ times.

DEFINITION 2. A matching of $UF(f_i)$ and $UF(f_j)$ is a 1-1 pairing of the elements of $UF(f_i)$ and $UF(f_j)$.

DEFINITION 3. The “match distance” ρ between f_i and f_j is the minimum sum of pairwise distances for all possible matchings of $UF(f_i)$ and $UF(f_j)$.

EXAMPLE. We will compute the match distance ρ for the following matrices, each having $M = 3$:

$$f_1 = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}; \quad f_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}; \quad f_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{bmatrix}.$$

For distance μ between the matrix entries (i, j) and (k, l) we use the city block distance $\mu\{(i, j), (k, l)\} = |i - k| + |j - l|$. The unfolded multisets are:

$$UF(f_1) = \{(1, 1), (1, 1), (2, 2)\}; \quad UF(f_2) = \{(1, 1), (2, 2), (3, 3)\}; \\ UF(f_3) = \{(2, 2), (3, 3), (3, 3)\}.$$

The elements $(1, 1)$ in $UF(f_1)$ and $(3, 3)$ in $UF(f_3)$ appear twice, since $f_1(1, 1) = 2$ and $f_3(3, 3) = 2$. The match distances using minimal matching of the unfolded sets are:

$$\rho(f_1, f_2) = \mu\{(1, 1), (1, 1)\} + \mu\{(1, 1), (2, 2)\} + \mu\{(2, 2), (3, 3)\} \\ = 0 + 2 + 2 = 4 \\ \rho(f_2, f_3) = \mu\{(1, 1), (2, 2)\} + \mu\{(2, 2), (3, 3)\} + \mu\{(3, 3), (3, 3)\} \\ = 2 + 2 + 0 = 4 \\ \rho(f_1, f_3) = \mu\{(1, 1), (2, 2)\} + \mu\{(1, 1), (3, 3)\} + \mu\{(2, 2), (3, 3)\} \\ = 2 + 4 + 2 = 8.$$

It can be easily verified that the above matchings are minimal.

As a comparison to vector distances, let

$$g_1 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}; \quad g_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}; \quad g_3 = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}.$$

Using vector distance all the g 's are equally far apart, but using the match distance yields $\rho(g_1, g_2) = \rho(g_2, g_3) = 2$ and $\rho(g_1, g_3) = 4$. This result is in better agreement with our intuitive notion of distance for these matrices.

THEOREM. *The match distance ρ is a metric, provided μ is a metric.*

Symmetry and positivity follow from the properties of μ as a metric. To prove the triangle inequality let $UF(f_1) = \{a_1, a_2, \dots, a_M\}$, $UF(f_2) = \{b_1, b_2, \dots, b_M\}$, and

$UF(f_3) = \{c_1, c_2, \dots, c_M\}$. Let π and σ be the permutations giving minimal matchings between f_1, f_2 and f_2, f_3 , respectively. Then

$$\begin{aligned} \rho(f_1, f_2) + \rho(f_2, f_3) &= \sum_i \mu(a_i, b_{\pi_i}) + \sum_i \mu(b_i, c_{\sigma_i}) \\ &= \sum_i \{ \mu(a_i, b_{\pi_i}) + \mu(b_{\pi_i}, c_{\sigma_{\pi_i}}) \} \\ &\geq \sum_i \mu(a_i, c_{\sigma_{\pi_i}}) \geq p(f_1, f_3). \end{aligned}$$

The first two equalities follow from the definitions of π and σ . The first inequality follows from the triangle inequality for μ , and the last inequality follows since ρ is the cost of a *minimal* matching.

The metric μ will generally depend on the problem domain. In the case of gray level histograms, for example, the distance between gray levels i and j might be $(i - j)^2, |i - j|$, etc. When the feature space L takes values from the indices of two-dimensional histograms, as in the case of co-occurrence matrices, μ can be taken as the city block, chessboard, or Euclidean distance. If L is a set of angles, μ can be defined as $\mu(\alpha, \beta) = \min\{|\alpha - \beta|, 2\pi - |\alpha - \beta|\}$.

3. COMPUTATIONAL COMPLEXITY

A heavy computational task when computing the match distance is to find the minimum matching between two multisets of order M . Although this problem appears at first glance to be combinatorial, there exist $O(M^3)$ algorithms for its solution [4]. For co-occurrence matrices of images of size $k \times k$, for example, we have a complexity of $O((k \times k)^3) = O(k^6)$.

In some cases, however, the minimal matching can be done even faster. In particular, when L is a subset of the real numbers, and μ is the absolute difference, the minimal matching can be computed very fast. This is the case when comparing two gray level histograms, and is also the case discussed in [3]. A minimal matching between a_1, a_2, \dots, a_M and b_1, b_2, \dots, b_M , when $\{a_i\}$ and $\{b_i\}$ are sorted, is simply $\langle a_1, b_1 \rangle, \langle a_2, b_2 \rangle, \dots, \langle a_M, b_M \rangle$. As a proof, notice that this claim is true for two multisets of size two. For larger sets, examine any two pairs $\langle a_i, b_{\pi_i} \rangle$ and $\langle a_{i+1}, b_{\pi(i+1)} \rangle$ in a minimal matching. Their distance $|a_i - b_{\pi_i}| + |a_{i+1} - b_{\pi(i+1)}|$ must be minimal as part of the minimal matching. If $b_{\pi_i} > b_{\pi(i+1)}$ we can exchange them without adding to the cost of the matching. Continuing in this way the matching will eventually become $\langle a_1, b_1 \rangle, \langle a_2, b_2 \rangle, \dots, \langle a_M, b_M \rangle$. If the multisets are initially sorted, the complexity of computing the match distance is $O(M)$, and it is $O(M \log M)$ otherwise. In most cases sorting both sets is not the only minimal matching. If the sets have some elements in common, any of them can be matched with the identical element in the other set, and the sorting could be carried out with the remaining elements. If there are many identical elements, substantial speedup might be gained.

Another interesting one dimensional problem arises in the case of cyclically ordered features. Such features were also extensively treated in [3], where the minimum was taken over all possible permutations. The authors of [3] used, as an approximation, only the cyclic permutations of the ordered multiset. We shall now prove that using these cyclic permutations is enough to find the minimum matching.

This follows from the following observations:

(a) Let $\{\langle a_i, b_{\sigma i} \rangle\}$ be a minimal matching. There exists a minimal matching in which no arc $\langle a_i, b_{\sigma i} \rangle$ includes another arc $\langle a_j, b_{\sigma j} \rangle$. If $\langle a_j, b_{\sigma j} \rangle \subseteq \langle a_i, b_{\sigma i} \rangle$, then replace these two pairs by $\langle a_i, b_{\sigma j} \rangle$ and $\langle a_j, b_{\sigma i} \rangle$ yielding the same cost. Graphically, we perform the substitutions as in Fig. 3.

(b) Using (a) to reduce the search space, and checking all possible combinations, it can be verified that a minimal matching for the case $M = 3$ is possible with a cyclically sorted permutation.

(c) In a minimal matching all triples $\langle a_i, b_{\sigma i} \rangle, \langle a_j, b_{\sigma j} \rangle, \langle a_k, b_{\sigma k} \rangle$ must also be minimal, and by (b) σ must be a cyclically sorted permutation on these three elements. This implies that the entire matching is a cyclically sorted permutation.

An even faster algorithm for matching of cyclically ordered features, with the same complexity as for linear features, appears in [8]. In the case where L is a set of Euclidean coordinates, and μ is the chessboard, city block, or Euclidean distance, approximation algorithms can be found with complexity between $O(M)$ and $O(M^2)$ [5, 6].

Several basic approximation algorithms exist, where the quality of the approximation depends on the specific application. One approach is to pass a space filling curve through the two matrices to be compared, the curve defining a mapping of the multidimensional histogram into a one-dimensional histogram. These one-dimensional histograms are then unfolded, and compared pointwise. Another approach is to use a “greedy” algorithm: we first pair all possible points in one matrix to points in the second matrix having the same coordinates. (This step corresponds to subtraction of one matrix from the other.) We then pair all points in one matrix with corresponding points at distance 1 in the second matrix, then all unpaired points in one matrix with unpaired points in the second matrix at distance 2, etc. A third approach is to recursively split the matrix into smaller regions [6]. When the regions are small enough, points within the regions are arbitrarily matched.

4. NORMALIZATION

Normalization is needed when the multisets to be compared are of different sizes. The problem is encountered, for example, when trying to compare the histograms of pictures of different sizes. Different approaches are possible; a few of these are described in this section.

A simple normalization is to duplicate all points in each set the same number of times, such that both sets have a number of elements equal to the lowest common multiple of the sizes of the two original sets. For example, to compare $A = (1, 0, 1)$ with $B = (2, 1, 0)$ we multiply A by 3 and B by 2 and compare $\hat{A} = (3, 0, 3)$ with $\hat{B} = (4, 2, 0)$. The latter have the same total number of points.

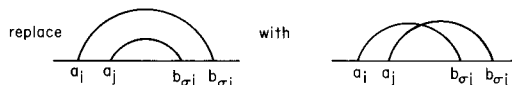


FIG. 3. Changing a pairing such that no arc will include another arc, without adding to the total arc length.

Another normalization method was used by Lavine *et al.* [14] for a point pattern matching algorithm that required an equal number of features. They padded the smaller feature set with random features having a distribution similar to the already existing features. This normalization method, however, results in a distance which is not a metric.

5. POSSIBLE APPLICATIONS

5.1. Texture Features

Co-occurrence matrices [1,2] are commonly used texture descriptors. In those matrices, given a translation δ , the (i, j) entry in the matrix is the number of pixel pairs at distance δ apart having the corresponding gray levels i and j . The match distance improves on the ordinary vector distance between co-occurrence matrices. In both methods, an initial subtraction removes identical elements. The vector distance just adds all difference elements after subtraction. In the match distance, subtraction is the matching of all identical objects in the two sets. Rather than just adding up the differences, the distances in a minimal matching between the positive and negative elements of the difference are used.

Consider, for example, addition of 1 to every gray level in a picture. In this case the co-occurrence matrix “slides” one point along the diagonal. The match distance between the two corresponding co-occurrence matrices will always be equal to the number of pixels. The vector distance, however, behaves differently for different pictures. For an initial picture with all gray levels even, for example, the distance between the co-occurrence matrices will be maximal: double the number of pixels. For other pictures the distance can be as small as double the square root of the number of pixels.

In another set of texture features [7], also represented by matrices, the match distance gave improved discrimination compared to the vector distance.

5.2 Shape Matching

In [9,10,11] Parui and Dutta Majumder suggest several methods for measuring shape and curve similarity. For shape dissimilarity, they suggest the mismatch area between the two shapes after size, position, and orientation normalization. When shapes are treated as planar point sets, the match distance can also be used as a similarity measure.

When looking for a minimal matching between two point sets, identical elements can always be paired together, and a minimal matching can be sought only for the remaining elements. In the context of shape matching, pairing identical elements leaves only the points in the mismatch area. Rather than simply counting the points in the mismatch area, the match distance weighs them by their distance in a minimal matching. As an example, consider matching the three shapes in Fig. 4. The mismatch area in this case is the number of mismatched points, and is 12 for both $d(A, B)$ and $d(A, C)$. For the match distance, however, the mismatched points are paired as in the figure, and the match distance is the sum of all pairwise distances. This gives $d(A, B) = 12$ and $d(A, C) = 24$ (when chessboard distance is used as the pairwise distance). The similarity of shape A to shape B is therefore revealed only by using the match-distance.

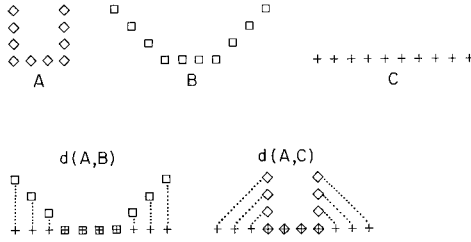


FIGURE 4.

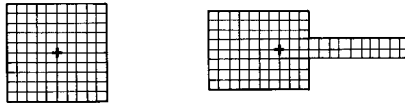


FIG. 5. Matching the two marked centroids yields a larger mismatch area than just matching the two lower-left corners. Centroid matching, however, does minimize the match-distance (see text).

Size, position, and orientation normalization are often used to bring two objects into maximum registration. The most commonly used method of normalization, registering the two centers of gravity, does not minimize the distance between the shapes (defined as the mismatch area) as desired. In Fig. 5, for example, the normalization that brings the two marked centroids together yields a total mismatch area of 64. With no position normalization, however, when the two lower-left corners coincide, the total mismatch area is only 40. This example shows the inadequacy of moment normalization for the shape distance using mismatch area.

Centroid normalization, however, does minimize the match distance between point sets, when pairwise distances are squared. Given two sets of M points $\{(x_i, y_i)\}, \{(\hat{x}_i, \hat{y}_i)\}$, and a matching that pairs every (x_i, y_i) with (\hat{x}_i, \hat{y}_i) , the distance between the sets under the translation t is

$$S(t) = \sum_i \left[((x_i + t_x) - \hat{x}_i)^2 + ((y_i + t_y) - \hat{y}_i)^2 \right].$$

Setting the derivatives to zero shows that the distance between the sets is minimized when the two centers of gravity are registered, regardless of the actual matching. Thus, the match distance is the most natural distance to use when centroid registration is performed.

5.3 Picture Comparison

As mentioned earlier, pictures themselves can be regarded as two-dimensional histograms when viewed as distributions of photons, where bright points indicate more photons than dark points.

Therefore, given two grey-level images, $P = \{p_{ij}\}, Q = \{q_{ij}\}$, where $\sum p_{ij} = \sum q_{ij}$, they can be compared using the match distance. When the above grey-level sum is not equal for the two pictures, they can be normalized using the methods of Section 4.

Trying to compare two pictures using the match distance is not always meaningful. A useful example, however, can be found in the problem of half-toning: printing grey-level pictures using black and white only. The quality of the half-toning can be defined using the match distance, as the need to have the same total "blackness" in both the grey-level and the binary images implies that the two sums of grey-level values are equal. Therefore, the black and white picture with minimal match distance to the original grey-level picture will be the best half-toning. The match distance is an improvement over measuring half-toning quality by human visual inspection.

6. CONCLUDING REMARKS

A distance metric has been described that can be used to compare matrices having equal sums of elements. These matrices were called "multidimensional histograms"; co-occurrence matrices are an example. This metric is applicable in several domains; in addition to co-occurrence matrices, applications to shape matching and picture half-toning were described.

This metric is shown to have many theoretical advantages over other methods, and in the half-toning case is the only existing quantitative measure. Unfortunately, computing the match distance is computationally expensive, and in some applications the added computation may not result in substantial improvement. However, when other comparison methods fail to give desired results, the match distance seems worth considering.

REFERENCES

1. R. M. Haralick, K. Shanmugam, and I. Dinstein, Textural features for image classification, *IEEE Trans. Systems Man Cybern.* **SMC-3**, 1973, 610–621.
2. L. S. Davis, S. A. Johns, and J. K. Aggarwal, Texture analysis using generalized co-occurrence matrices, *IEEE Trans. Pattern Anal. Mach. Intell.* **PAMI-1**, 1979, 251–259.
3. H. C. Shen and A. K. C. Wong, Generalized texture representation and metric, *Comput. Vision Graphics Image Process.* **23**, 1983, 187–206.
4. N. Christofides, *Graph Theory: An Algorithmic Approach*, Academic Press, London, 1975.
5. M. E. Dyer and A. M. Frieze, A partitioning algorithm for minimum weighted Euclidean matching, *Information Processing Letters* **18**, 1984, 59–62.
6. K. Y. Supowit and E. M. Reingold, Divide and conquer heuristics for minimum weighted Euclidean matching, *SIAM J. Comput.* **12**, 1983, 118–143.
7. M. Werman and S. Peleg, Multiresolution texture signatures using min–max operators, in *Proc. 7th Intl. Conf. on Pattern Recognition*, 1984, pp. 97–99.
8. M. Werman, S. Peleg, R. Melter, and T. Y. Kong, Bipartite graph matching for points on a line or a circle, *J. Algorithms*, in press.
9. S. K. Parui and D. Dutta Majumder, A new definition of shape similarity, *Pattern Recognit. Lett.* **1**, 1982, 37–42.
10. S. K. Parui and D. Dutta Majumder, Shape similarity measures for open curves, *Pattern Recognit. Lett.* **1**, 1982, 129–134.
11. S. K. Parui and D. Dutta Majumder, How to quantify shape distance for 2-dimensional regions, in *Proc. 7th Intl. Conf. on Pattern Recognition*, 1984, pp. 72–74.
12. A. Rosenfeld and A. Kak, *Digital Picture Processing*, 2nd ed., Academic Press, New York 1981.
13. C. E. Kim and T. A. Anderson, Digital disks and a digital compactness measure, in *Proc. 7th Intl. Conf. on Pattern Recognition*, 1984, pp. 254–257.
14. D. Lavine, B. A. Lambird, and L. N. Kanal, Recognition of spatial point patterns, *Pattern Recognit.* **16**, 1983, 167–182.
15. R. Hummel, Image enhancement by histogram transformation, *Comput. Graphics Image Process.* **6**, 1977, 184–195.