Automatic Matching of Buildings and Corners

Han-Wen Hsiao and Kam W. Wong

Abstract

To update a portion of an existing cartographic database, the common practice is to relate a new data file to an existing file by means of survey control points that are included in both files. In the absence of such survey control points, well-defined points such as building corners can be used. This paper presents an algorithm to perform matching of common buildings and building corners in vector data files. The algorithm starts with a Fourier-based initial matching. A sequence of validity checks combined with robust estimation provides a complete recognition of common buildings. Matching of individual corner points is performed by using a similarity parameter, followed by a series of checks and validations. The two maps may have different scales, different coordinate systems, and no identifying cartographic labels. Experimental results have demonstrated the robustness of the algorithm.

Introduction

Digital cartographic data files in vector format are now commonly used in all aspects of engineering: design, planning, construction, and operation and maintenance. With the rapid development of modern societies, there is a real need for algorithms that can facilitate the automatic updating of existing databases. To update a portion of an existing cartographic database, the common practice is to relate the new data file to the existing file by means of survey control points that are included in both files. However, such survey control points are often not available, and matching of corresponding points between the two files is often performed manually. This paper presents a matching algorithm for automatic updating of common building features without *a priori* information of corresponding points in two cartographic files.

Consider, for example, the two files graphically represented in Figures 1 and 2. The file in Figure 1 has an original scale of 1:2,500 and may be considered to be a part of an archive database for an entire city. The file in Figure 2 has an original scale of 1:500, and is a CAD file that has been generated for the redevelopment of a neighborhood. There are recognizable buildings that are common in both files, although differences may exist in the details of the corresponding buildings in the two files. The CAD file is more current in time. It can be seen that new buildings have been added, an old building has been removed, and some old buildings have been expanded. There are no known homologous points anywhere within the area. The purpose of the algorithm to be presented here is to automatically match building corners in the two files. The process consists of two major steps: (1) recognition of common buildings, and (2) matching corners in the identified common buildings.

For the purpose of generality, the algorithm assumes that the two data files contain only strings of coordinates (x, y) representing each building, with no identifying information for any of the buildings. The string of coordinates for a building must form a closed polygon, but the building corners can be arranged in either a clockwise or a counterclockwise direction. The starting point for the sequence of corner points representing a building can be completely random, and be different between the two files. As is the case commonly encountered in practice, the same building in each of the two CAD files may also consist of a different number of corners due to cartographic generalization or interpretation. The two files can also have different scales and different coordinate systems.

Matching Buildings

Shape analysis is an important phase of pattern recognition, and many techniques can be found in the literature relating to this issue: B-splines (Jain, 1989), autoregressive models (Jain, 1989; Kauppinen *et al.*, 1995), Fourier descriptors (Zahn and Roskies, 1972; Richard and Hemami, 1974; Persoon and Fu, 1977; Wallace and Wintz, 1980; Proffitt, 1982; Jain, 1989; Arbter *et al.*, 1990; Kauppinen *et al.*, 1995; Rothe *et al.*, 1996; Tseng *et al.*, 1997), etc. In general, the Fourier-based methods using different models for boundary representation provide superior performance in most cases (Kauppinen *et al.*, 1995). The algorithm to be presented in this paper is also based on the use of Fourier descriptors for initial matching. The innovation of the algorithm lies in the development of multiple levels of checking procedures without normalization of rotation and starting point to solve the specific problems relating to the recognition of common buildings in cartographic data files. The algorithm consists of the following main steps:

(1) Rearrangement of all data points within each polygon into a counterclockwise sequence;
(2) Computation of the Fourier descriptors of each polygon;
(3) Matching of polygons in the two files by cross correlation; and
(4) Performance of three separate, but increasingly rigorous, validity checks of matched polygons:
   - by size-distance ratios,
   - by conformal transformation with robust estimation, and
   - by geometric overlay after coordinate transformation.

These steps will each be discussed in details in the following paragraphs.

Counterclockwise Sequence

The sequence of points within each polygon in the two data files is checked first by calculating the unit scalar ξ_j at each corner point j: i.e.,

$$\xi_j = \mathrm{Sgn}[(V_j - V_{j-1}) \times (V_{j+1} - V_j)] \quad (1)$$

where the vector V_j of point j has three components $(x_j, y_j, 0)$ and $(V_j - V_{j-1}) \times (V_{j+1} - V_j)$ is the cross product of the vectors $(V_j - V_{j-1})$ and $(V_{j+1} - V_j)$. It can be seen from Figure 3

Photogrammetric Engineering & Remote Sensing,
Vol. 65, No. 7, July 1999, pp. 803–810.

0099-1112/99/6507–803$3.00/0

Department of Civil Engineering, University of Illinois at Urbana-Champaign, 205 N. Mathews Avenue, Urbana, IL 61801 (h-hsiao@uiuc.edu; k-wong1@uiuc.edu).

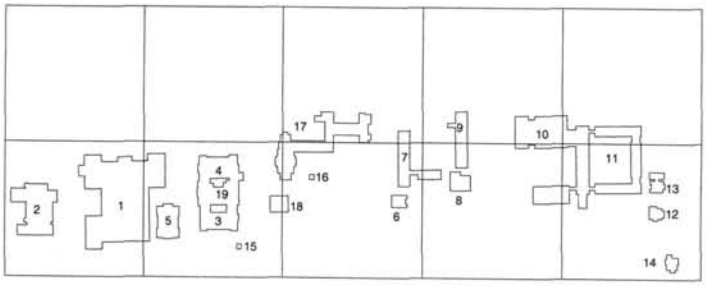

Figure 1. Archive map of Case I (original scale 1:2,500).

that, when the corner points are arranged in a counterclockwise sequence, the vertex j has a positive (negative) value of ξ_j in the z direction and is a convex (concave) corner of the building. For a clockwise sequence, the sum of the ξ_j values for all the corners forming a building will be negative. Such a sequence is rearranged into a counterclockwise sequence.

Fourier Descriptors of Polygons

The next step is to compute the Fourier descriptors to extract useful attributes for each polygon. Two different procedures for computing the Fourier descriptors of polygons are commonly used: cumulative angular function (Zahn and Roskies, 1972) and complex coordinate function (Richard and Hemami, 1974; Persoon and Fu, 1977; Wallace and Wintz, 1980; Proffitt, 1982). The latter was selected in this study for its simplicity. Equation 2 is the formula for computing the Fourier descriptors of a polygon (Persoon and Fu, 1977). The $(b_{k-1} - b_k)$ term is actually the curvature vector of a corner, and will also be saved as an attribute for point k.

$$D_n = \frac{L}{(2\pi n)^2} \sum_{k=1}^{N} (b_{k-1} - b_k) \exp(-j2\pi n l_k / L) \tag{2}$$

where $l_k = \sum_{i=1}^{k} |V_i - V_{i-1}|$ for $k > 0$ and $l_0 = 0$,

$$b_k = \frac{V_{k+1} - V_k}{|V_{k+1} - V_k|},$$

$n = 1, ..., N - 1$, and

N = number of corners in a polygon.

The D_0 term, which is not included in the above equation, represents the centroid of a polygon and is obtained by computing the center of mass (Prisley *et al.*, 1989) of the polygon. Although the D_0 term will be set to zero to normalize the translation of the polygons, it is an important attribute and is saved for use in later steps.

Because two corresponding buildings may differ in position, orientation, size, starting point, and a certain degree of dissimilarity, the computed Fourier descriptors must be normalized prior to cross correlation. Unlike the normalization algorithms in the literature (Zahn and Roskies, 1972; Richard and Hemami, 1974; Wallace and Wintz, 1980; Proffitt, 1982; Arbter *et al.*, 1990; Rothe *et al.*, 1996), only the removals of translation and scale factors are considered, because two maps are usually subject to conformal transformation. Assigning zero to the D_0 term normalizes the translation factor. The size in the frequency domain is normalized by dividing the Fourier descriptors by the square root of the summation of the power spectrum (Proffitt, 1982), i.e., the mean quadratic distance of all points to the center of gravity

$$D_n^* = D_n \Big/ \sqrt{\sum_{n=1}^{N-1} |D_n|^2} \tag{3}$$

for $n = 1, ..., N - 1$. After normalization of translation and size, the center of the building lies at the origin and the norm of the new Fourier descriptors is equal to one. The mean quadratic distance will be used later as a size factor σ for performing the validity check.

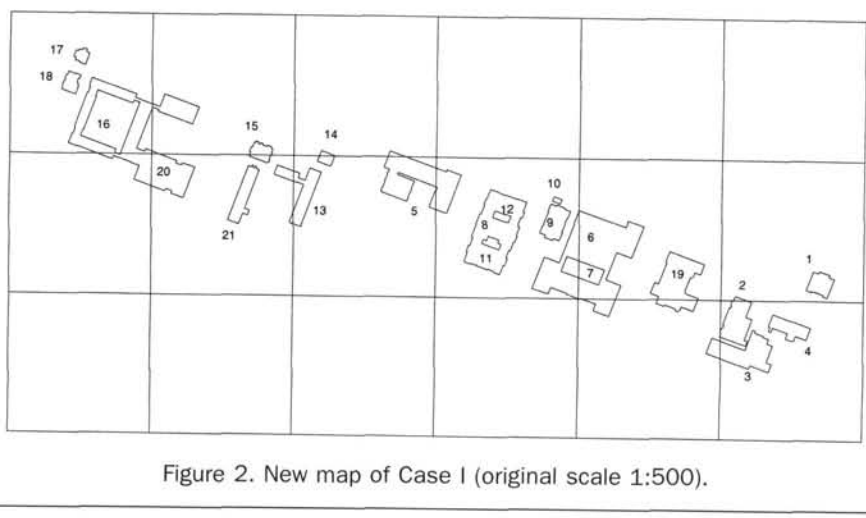

Figure 2. New map of Case I (original scale 1:500).

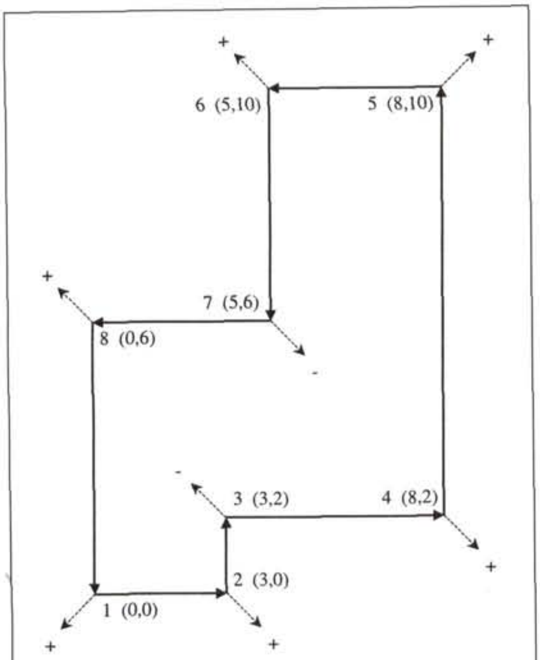

Figure 3. Unit scalar ξ_j and curvature vector $(b_{k-1} - b_k)$.

Matching by Cross Correlation

The method of cross correlation is used to identify the most likely matches of corresponding buildings in the two files. Let p_i represent the Fourier descriptors of a building in the archive file, and q_j represent the Fourier descriptors of a building in the new file; where $i = 0, ..., M - 1$; $j = 0, ..., N - 1$; and $M > N$. The cross correlation is computed as

$$c_g = \sum_{k=0}^{M-1} p_k \bar{q}_k \exp(-j2\pi kg/M) = \sum_{k=0}^{M-1} u_k \exp(-j2\pi kg/M) \quad (4)$$

where $g = 0, ..., M - 1$. The cross correlation is equivalent to performing the discrete Fourier transform of u_k, which is the multiplication of p_k and the complex conjugate of q_k (Richard, 1974), and $u_N, ..., u_{M-1}$ are padded zero (Proffitt, 1982). The similarity measure S_{pq} of pair (p, q) is then obtained as

$$S_{pq} = \max_{g=0}^{M-1} |c_g| \quad (5)$$

and does not deal with the normalization of the starting point. If there were m buildings in the archive file and n buildings in the new file, there would be $m \times n$ possible pairs of matches. The similarity measures of the $m \times n$ possible matches are computed to construct a similarity table for matching buildings. A pair of buildings is considered a preliminary match if its similarity measure S_{pq} is the maximum along both the row and column in the similarity table (Hu and Ahuja, 1994). Let $\Im$ be a set of matched pairs. A pair P_{pq} is included in this set if it meets the following criterion:

$$P_{pq} \in \Im, \quad \text{if} \max_{i=1}^{m}(S_{iq}) = \max_{j=1}^{n}(S_{pj}) \quad (6)$$

where $1 \le p \le m$ and $1 \le q \le n$. Setting a threshold for the similarity measure S_{pq} is discouraged, because finding an appropriate value is unreasonable and difficult. The validity is examined by the geometry of the centroid pattern.

Size-Distance Ratio Check

Because matched pairs in the set $\Im$ are similar in terms of normalized Fourier descriptors, two corresponding buildings might not be matched if the difference between them is significant. On the other hand, two entirely different buildings might be matched as a pair when their normalized Fourier descriptors are very similar. Such circumstances happened frequently when two maps have very different map scales. Incorrect matches could be identified by some validity checks. A clustering technique (Stockman, 1987) can be employed to determine the scale between two corresponding objects. Solving for the transformation parameters, however, is not necessary during the stage of building matching. Moreover, using the edge segments of all buildings for matching would increase the computation burden. Matching buildings at this point is regarded as point pattern matching using the centroids, but the size information is also used to simplify the problem. The validity of the matched pairs obtained from cross correlation is first checked by comparing the relative size and separation distances between two matched pairs. It is noticed that the centroids are derived from the building polygons and are subject to variation due to cartographic generalization, interpretation, actual remodeling, and/or other error sources. Therefore, matching the centroids by clustering in a parameter space had been found through testing to be an inappropriate method.

As shown in Figure 4, let σ_i^A and σ_i^N denote the size attribute, represented as the radius of a circle, of buildings in the archive and new files, respectively. These two buildings have been matched and saved as pair i in the set $\Im$ contain-

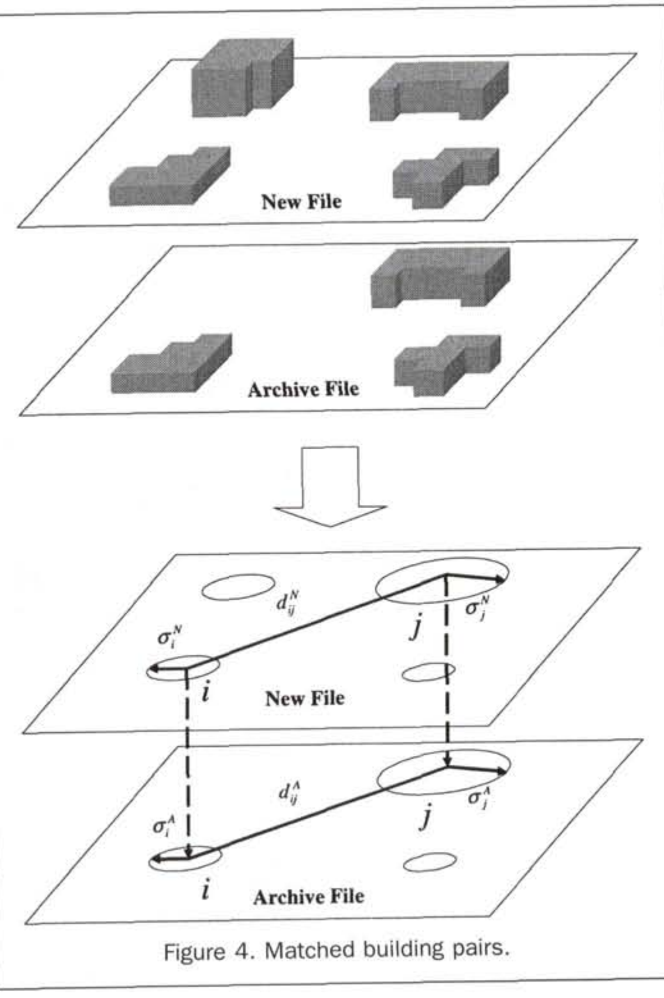

Figure 4. Matched building pairs.

ing t matches. Furthermore, let d_{ij}^A and d_{ij}^N represent the distances between the centroids of buildings i and j in the archive and new files, respectively. To investigate whether pair i is a correct match, the other temporarily matched pairs should give supports to pair i, and the best support should meet the following criterion:

$$\min_{\substack{j=1\\ j\neq i}}^{t}\left\{\max\left(\left|\frac{r_{\sigma_j}}{r_{\sigma_i}}-1\right|,\left|\frac{r_{\sigma_i}}{r_{d_{ij}}}-1\right|\right)\right\}<0.1 \qquad (7)$$

where $r_{\sigma_i}=\sigma_i^N/\sigma_i^A$ and $r_{d_{ij}}=d_{ij}^N/d_{ij}^A$.

The size ratio alone is insufficient to identify incorrect matches. Including the distance ratio provides the ability to examine the geometry of centroid distribution. If pairs i and j are correct matches, then two size ratios and one distance ratio should consistently reflect the scale. It is better to normalize the ratios so that the criterion is independent of data sets. The design of Equation 7 forces the computed value of a correct pair to be very close to zero. Defining the threshold is crucial to filter out mismatched pairs. It is not expected that the difference in distances between two buildings on two maps would exceed ten percent of the actual distance. The robustness of the size-distance ratio check lies in its ease to distinguish the correct pairs from the mismatched ones. It should be noted that this procedure requires that there be at least two correctly matched pairs in the set $\mathfrak{I}$. Although the derived centroids are not as accurate as survey control points to compute transformation parameters, they are insensitive enough for identification of incorrect pairs.

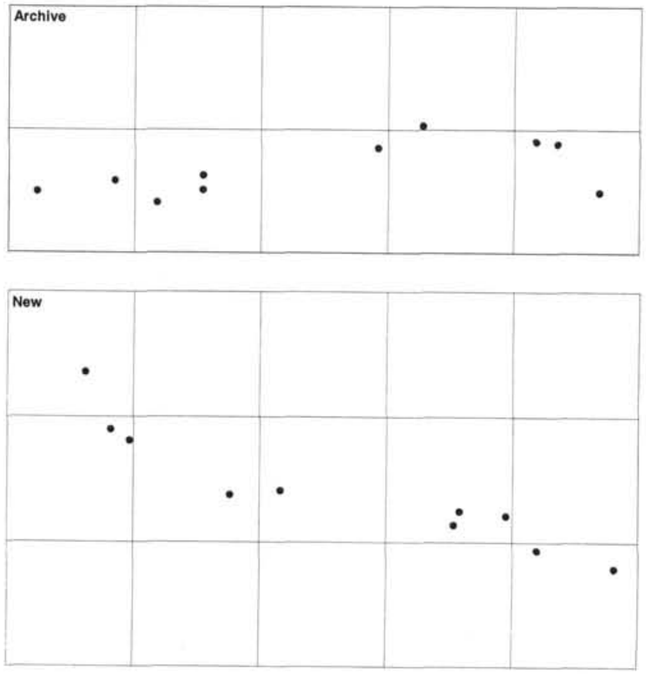

Figure 5. Location of matched building centroids.

Conformal Transformation with Robust Estimation

A second validity check of the remaining matched pairs is conducted by performing a conformal transformation of the new file into the archive file. Incorrect matches are identified in the process by robust estimation. Figure 5 shows the centroid locations of the remaining matched pairs in the two files. The conformal transformation parameters between the two coordinate systems are computed using the centroids of the corresponding buildings in the two files. Any incorrectly matched pair is identified by the large residuals in the corresponding centroid coordinates in the least-squares solution. The following weighting function (Krarup *et al.*, 1980; Chen and Lee, 1992) for the coordinates has been found to be effective in filtering out the incorrectly matched pairs:

$$w_i=\begin{cases}1 & \text{when } |v_i|<2\hat{\sigma}_0\\ \exp\left[-0.05\left(\frac{|v_i|}{\hat{\sigma}_0}\right)^{4.4}\right] & \text{for iterations 1,2,3 when } |v_i|>2\hat{\sigma}_0\\ \exp\left[-0.05\left(\frac{|v_i|}{\hat{\sigma}_0}\right)^{3.0}\right] & \text{for iteration 4 and afterwards}\end{cases} \qquad (8)$$

Although the size-distance check can usually filter out most mismatched pairs, this step is not redundant at all. In addition to providing an additional check on the match results, this step also provides preliminary values for the coordinate transformation parameters for the new file (Besl and McKay, 1992).

Geometric Overlay after Coordinate Transformation

After coordinates of the new file have been transformed into the same system as the archive file, a geometric overlay analysis is performed to serve three purposes: (1) verify the validity of the matched polygons, (2) identify any additional matches that have been missed by the previous steps, and (3) organize the matched buildings into match groups. In large-scale maps, buildings can be represented by polygons of rather complex shape. Sometimes several separate structures are represented as a single polygon. It is also possible that a single building is actually represented by more than one polygon. These situations often occur when buildings have courtyard, patio, or attached structures, or have undergone remodeling. Table 1 illustrates how the groups of matched buildings are automatically organized. For example, Groups 0 consists of buildings that are found in only one of the two files, and buildings that have been mismatched. Group 1 contains only building No. 1 in the archive file, but the building has been remodeled. In the new file, the additional polygon No. 7 must come with the original building No. 6 through the individual group transformation. Group 2 consists of one matched pair of buildings: building No. 2 in the archive file, and building No. 19 in the new file. Group 3 shows that building No. 3 in the archive overlaps with buildings Nos. 8 and 12 in the new file; building No. 4 overlaps with buildings Nos. 8 and 11 in the new file; and building No. 19 overlaps with buildings Nos. 8, 11, and 12 in the new file. The missing Groups 7 and 11 are thus categorized as mismatched pairs collected in Group 0. It can be seen that organizing the matched pairs into different groups is highly complex, because it requires recursively searching the archive and new feature ID numbers. However, these matched pairs have been stored in the database, and the task is achieved by one standard SQL query. At the end of this step, the task of matching common buildings from the two files is complete.

Matching Corners

The matching of common building corners between the archive file and the new file is performed among buildings with each match group. Although a group may consist of more than one building, the sequence of building corners that form a closed polygon is maintained for each individual building within a group. The matching process may be summarized as follows:

- Perform preliminary matching by means of a similarity parameter.

Table 1. Grouped Polygons

Group ID	Archive	New	Comment
0	0	1	New
	0	2	
	0	3	
	0	4	
	0	10	
	14	0	Old
	15	0	
	16	0	
	17	0	
	8	15	Mismatch
	13	18	
1	1	6	One to many
	1	7	
2	2	19	One to one
3	3	8	Many to many
	3	12	
	4	8	
	4	11	
	19	8	
	19	11	
	19	12	
4	5	9	One to one
5	6	14	One to one
6	7	13	One to one
8	9	21	One to one
9	10	16	Many to many
	10	20	
	11	16	
	11	20	
10	12	17	One to one
12	18	5	One to one

- Perform validity check according to sequence order of points within each building.
- Perform validity check according to estimated accuracy of data points.
- Perform conformal transformation with robust estimation to identify remaining mismatched points.
- Points from both files are combined and merged to form closed polygon, representing the building corners in the updated file.
- Cartographic rendering is performed to square the building corners.

Corner Matching by Similarity Parameter

Let s_{ij} represent the similarity parameter between corner i in an archive group of buildings and corner j in the matching group of buildings in the new file. It is computed by the following expression:

$$s_{ij} = \frac{(b_{i-1} - b_i) \cdot (b_{j-1} - b_j)}{1 + D_{ij}} \tag{9}$$

where $(b_{i-1} - b_i)$ is the curvature vector of corner i computed from Equation 2, $(b_{j-1} - b_j)$ is the curvature vector of corner j computed from Equation 2, and D_{ij} is the distance between corners i and j in the archive coordinate system. The similarity parameter provides a measure of similarity of the curvature of the two corners, and the closeness of the two corners.

Suppose that there are m corners in the building group in the archive file, and n corners in the matching group in the new map. Totally, $m \times n$ values of S_{ij} are computed to construct a similarity table for all possible combinations of matching corners between the two groups. Define $\aleph$ as a collection of matched pairs of building corners. Pair p_{ij} is included in the collection if it satisfies the following criterion:

$$p_{ij} \in \aleph, \quad \text{if} \quad \max_{k=1}^{m}(s_{kj}) = \max_{l=1}^{n}(s_{il}) \quad \text{and} \quad \xi_i \xi_j > 0 \tag{10}$$

where $1 \le i \le m$ and $1 \le j \le n$. Even if two matched buildings in the two files are identical in shape, the building corners may still have some differences in coordinates due to the preliminary transformation of the new map using the centroids of buildings as described previously. Unless the new map is exactly transformed into the coordinate system of the archive map, assigning a threshold value to the distance D_{ij} is of no meaning. The merit of Equation 10 lies in eliminating the need for threshold values of s_{ij} and D_{ij}. The $\xi_i \xi_j > 0$ term is used to distinguish convex/concave corners.

Validity Check According to Point Order

Figure 6 shows the matched corner points between two buildings derived from similarity analysis as described above. Because of differences in the cartographic details between the two maps, corners Nos. 159, 166, 167, and 168 in the archive file are mistakenly matched to corners Nos. 160, 161, 152, and 159 respectively in the new file. These mismatches can be identified by means of the sequence order of points within the original polygon in the new file.

First, matched pairs of corner points are arranged in a counterclockwise sequence in the order of the archive file, as shown in Figure 6. The unit scalar ξ_j is computed for each matched point in both the archive and new file using Equation 1. The two points in a matched pair should have identical values of ξ_j, i.e., both -1 or both $+1$. For the example in Figure 6, the two pairs (167, 152) and (168, 159) can thus be identified as false matches. After eliminating false matches in this manner, a search is made through the sequence of corner points for the new file. Logically, at corner point j, either the next point $(j + 1)$ or the previous point $(j - 1)$ should be in proper sequence order. If neither is in order, then the match pair including point j in the new file is also eliminated from the list of matched pairs. For the example in Fig-

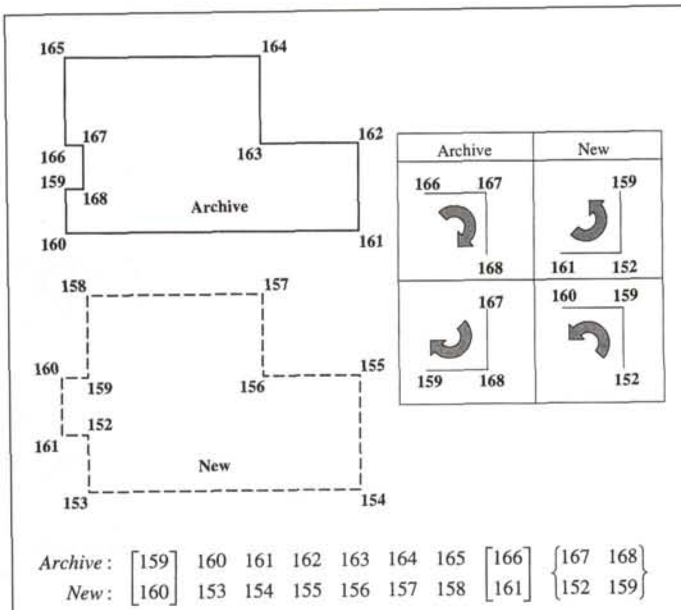

Figure 6. Validity check according to point order.

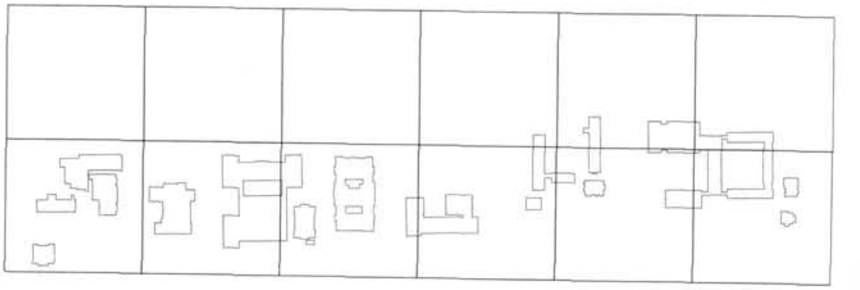

Figure 7. Merge map of Case I.

ure 6, after the elimination of two matched pairs from the list, the remaining pairs are arranged in sequence as follows:

Archive:	**159**	160	161	162	163	164	165	**166**
New:	**160**	153	154	155	156	157	158	**161**

At point No. 154 in the new file, the next point along the list of matched pairs is No. 155 and the previous point is No. 153. A comparison with the original order in the new file (see Figure 6) would indicate that the order is correct in both directions. At point No. 160, the next point should be No. 153, and the previous point should be 161 (wrapped around); both of which are incorrect and therefore the pair (159, 160) is a false match. Similarly, the pair (166, 161) can also be identified as a false match in this manner.

Distance Check

A second validity check of the matched pairs of corners is performed using *a priori* knowledge about the positional accuracy of the two maps. Based on the map scales and the National Map Accuracy Standards, the estimated standard error of the coordinates in both the archive and the new maps are computed. Let p_{ii^*} denote the pair of matched corners that include corner i in the archive file, and corresponding corner i^* in the new file. The following criterion must be satisfied if it is considered a correct pair:

$$\min_{\substack{j=1 \\ j\neq i}}^{t} (d_{ij^*}) < \max(\sigma_i, \sigma_{i^*}) + \max(\sigma_j, \sigma_{j^*})$$

where

$$d_{ij^*} = \sqrt{(x_{ij} - x_{i^*j^*})^2 + (y_{ij} - y_{i^*j^*})^2}, \quad (11)$$
$$x_{ij} = x_i - x_j,\ x_{i^*j^*} = x_{i^*} - x_{j^*},$$
$$y_{ij} = y_i - y_j,\ y_{i^*j^*} = y_{i^*} - y_{j^*}, \text{ and}$$
t = number of matched pairs within group.

Conformal Transformation with Robust Estimation

Let $\aleph$ denote the remaining matched pairs of corner points for two building groups. A conformal transformation of the coordinates from the new file to the archive file is performed with robust estimation to further eliminate poor matches. A stable solution is usually obtained after three iterations.

Merging of Information from Two Files

The process of merging information from the new file into the archive file involves the following steps:

(1) Compute more accurate values for the conformal transformation parameters by using the coordinates of the matched corner points.
(2) Transform the point coordinates of the new file into the coordinate system of the archive file. Estimated standard errors of the transformed coordinates are also computed by means of error propagation.
(3) Points from both files are combined and merged to form closed polygons.
(4) Finally, cartographic rendering is performed to square the building corners.

Test Results

Figure 7 shows the file resulting from matching and merging the new file in Figure 2 to the archive file in Figure 1. This case included three buildings with inside courtyards, represented by smaller polygons enclosed within the buildings. In this case, the scale of the archive map is smaller than that of the new map. The resulting matched file showed that the algorithm was effective in accommodating these situations.

Figures 8a, 8b, and 8c show the archive, new, and merged files of a second test case. Most of the buildings in this case had unique shapes, and the Fourier descriptors were highly effective in providing the correct matches. Building No. 3 in the archive map in Figure 8a had been extensively remodeled, with a large part of the original building demolished. The remnants of this buildings were represented as two buildings (Nos. 14 and 15 in Figure 8b) in the new map. Corner matching was successfully accomplished. Because the new files in Case I and II were more current and had a larger scale, i.e., more details and higher positional accuracy, the merged map looked essentially like the new map. However, it differed from the original new map in Figure 8b in that it had been transformed into the archive coordinate system and the precision information of points had been updated.

Figures 9a, 9b, and 9c show the archive, new, and merged files of a third test case which included several buildings of similar shape (rectangular) but different sizes. The normalized Fourier descriptors could not distinguish them, and had to rely on the size-distance ratio check to identify the false matches. In order to demonstrate the ability of the algorithm to merge small-scale maps into archive maps of larger scale, the archive map in this case was assigned a scale of 1:500, while the new map had a scale of 1:2,500. In such a circumstance, the merging strategy was still based on the feature in the new map because the information was more current. Any details that appeared in the archive map but did not show in the new map would be retained if the details were invisible in the smaller scale of the new map, such as building 1 in the archive map and building 2 in the new map. Otherwise, the details would be treated as being remodeled and removed from the archive map, for example, building 20 in the archive map and building 22 in the new map.

Conclusions

An algorithm has been successfully developed for the automatic recognition of common buildings and building

Figure 8. (a) Archive map of Case II (original scale 1:2,500). (b) New map of Case II (original scale 1:500). (c) Merge map of Case II.

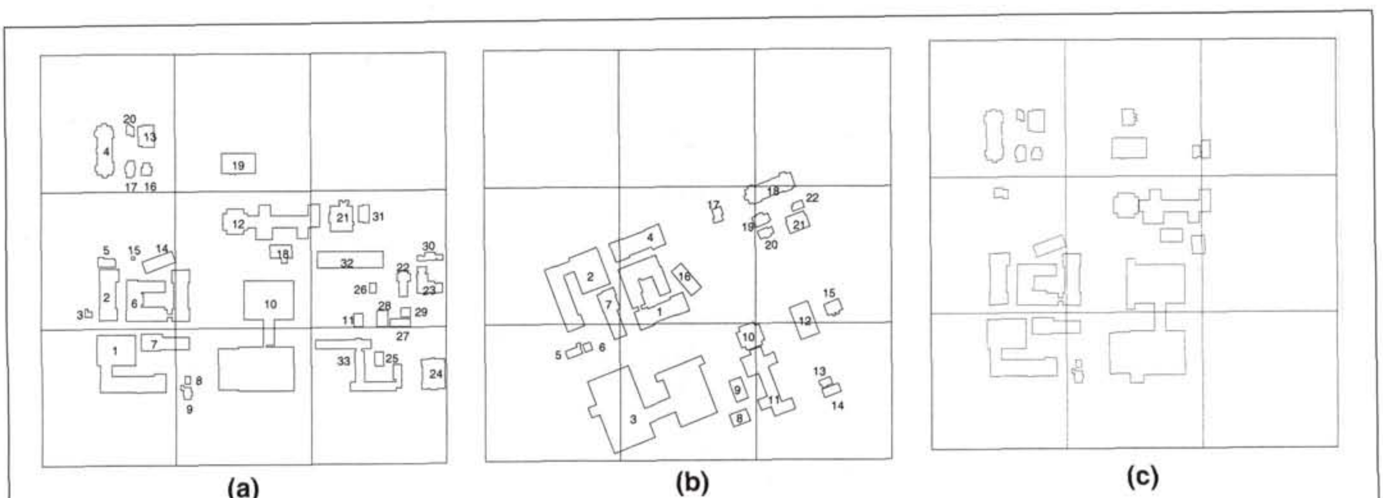

Figure 9. (a) Archive map of Case III (original scale 1:500). (b) New map of Case III (original scale 1:2,500). (c) Merge map of Case III.

corners from two cartographic data files, and for merging the information contents of the two files. The algorithm has been found to be robust, and computationally efficient. For the three reported test cases, the entire process of recognizing common buildings and merging the two files was accomplished automatically in less than 2 minutes of time using a 166-MHz desktop personal computer. The algorithm does not require any predetermined threshold values. However, it does require that there be at least two common buildings in the two data files. As a by-product of the merging process, estimated standard errors are computed by error propagation for all the position coordinates in the merged files. Although the algorithm has been developed specifically for the automatic updating of existing cartographic databases, it also has potential applications in the matching of common features in stereo pairs of photographs, for change detection in remote sensing, and for accuracy verification of new map files.

References

Arbter, K., W.E. Snyder, H. Burkhardt, and G. Hirzinger, 1990. Application of affine-invariant Fourier descriptors to recognition of 3-D objects, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):640–647.

Besl, P.J., and N.D. McKay, 1992. A method for registration of 3-D shapes, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256.

Chen, L.C., and L.H. Lee, 1992. Progressive generation of control frameworks for image registration, *Photogrammetric Engineering & Remote Sensing*, 58(9):1321–1328.

Hu, X.P., and N. Ahuja, 1994. Feature extraction and matching as signal detection, *International Journal of Pattern Recognition and Artificial Intelligence*, 8(6):1343–1379.

Jain, A.K., 1989. *Fundamentals of Digital Image Processing*, Prentice-Hall, Englewood Cliffs, New Jersey, 569 p.

Kauppinen, H., T. Seppänen, and M. Pietikäinen, 1995. An experimental comparison of autoregressive and Fourier-based descriptors in 2D shape classification, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(2):201–207.

Krarup, T., J. Juhl, and K. Kubik, 1980. Götterdämmerung over least squares adjustment, *The 14th Congress of the International Society of Photogrammetry, Commission III*, B3:369–378.

Persoon, E., and K.S. Fu, 1977. Shape discrimination using Fourier descriptors, *IEEE Transactions on Systems, Man, and Cybernetics*, 7(3):170–179.

Prisley, S.P., T.G. Gregoire, and J.L. Smith, 1989. The mean and variance of area estimates computed in an arc-node geographic information system, *Photogrammetric Engineering & Remote Sensing*, 55(11):1601–1612.

Proffitt, D., 1982. Normalization of discrete planar objects, *Pattern Recognition*, 15(3):137–143.

Richard, C.W., and H. Hemami, 1974. Identification of three-dimensional objects using Fourier descriptors of the boundary curve, *IEEE Transactions on Systems, Man, and Cybernetics*, 4(4):371–378.

Rothe, I., H. Süsse, and K. Voss, 1996. The method of normalization to determine invariants, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(4):366–376.

Stockman, G., 1987. Object recognition and localization via pose clustering, *Computer Vision, Graphics, and Image Processing*, 40:361–387.

Tseng, Y.H., J.J. Tzen, K.P. Tang, and S.H. Lin, 1997. Image-to-image registration by matching area features using Fourier descriptors and neural networks, *Photogrammetric Engineering & Remote Sensing*, 63(8):975–983.

Wallace, T.P., and P.A. Wintz, 1980. An efficient three-dimensional aircraft recognition algorithm using normalized Fourier descriptors, *Computer Graphics and Image Processing*, 13:99–126.

Zahn, C.T., and R.Z. Roskies, 1972. Fourier descriptors for plane closed curves, *IEEE Transactions on Computers*, 21(3):269–281.