BUREAU OF THE CENSUS
STATISTICAL RESEARCH DIVISION REPORT SERIES
SRD Research Report Number: Census/SRD/RR-87/24

Conflation: Automated Map Compilation

by

Alan Saalfeld
Statistical Research Division
Bureau of the Census

Recommend by: Tom O'Reagan

Report completed: September, 1986

Report issued: October 1, 1987

CONFLATION: AUTOMATED MAP COMPILATION

Alan Saalfeld

Statistical Research Division
Bureau of the Census
Department of Commerce, USA

CONFLATION MOTIVATION AND HISTORY

Origins

Although it was contemplated several years earlier [18], conflation, or automated map compilation, first became a reality in 1985 [5] through a fortuitous combination of circumstances. First, there was a clear need for some computerized system for doing a large amount of compilation of digital map data. That need arose within a joint project between the United States Geological Survey and the Bureau of the Census [17] to consolidate their separate digital map files of metropolitan areas of the United States, consisting of over 5,700 pairs of metropolitan map sheet files. The magnitude of the task justified a major investment of computer and human resources to develop a computerized system. The size of the task also recommended automating the system as fully as possible.

Second, the technology of graphics terminals was sufficiently advanced to support the real-time interactive map image manipulation that proved to be indispensable for developing, testing, and executing conflation system subroutines. Many of the models for conflation system modules were able to be built and tested on large data sets with excellent response times due to recent hardware advances for drawing and re-drawing images. Those same models could not have been implemented two years earlier.

Third, the rapid development and implementation of new mathematical algorithms, especially in the relatively new area of computational geometry [7], permitted the major software development necessary for a successful conflation system. The sound underlying mathematical map model in use at the Bureau of the Census also facilitated the application of new computational techniques to the problem of automated compilation. The conflation system fostered, and then used, innovative triangulation routines [2, 10, 12], topological transformations [3, 11], statistical matching methods [14], and pattern recognition techniques [9, 13]. Some of these conflation system components are now being used in other applications [14, 15].

Developing a conflation system required a special blend of mathematical and computational theory, programming staff and computer hardware capable of implementing that theory, and a real map compilation problem large enough to justify a major investment of resources. The Census Bureau's prototype system required approximately two person-years for its development by one research mathematician and two applications computer scientists. Further development of the prototype and implementation of a production conflation system have been postponed to permit staff to work on high priority activities related to the decennial census.

Evolution

By demonstrating feasibility and by providing theory for others to build upon, the initial work at the Bureau of the Census prompted interest and activity in conflation outside the Bureau. Engineering firms [1], public utilities, and developers of geographic information systems [4] began implementing their own versions of conflation system modules similar to the models provided by the Bureau of the Census.

One of the motivating factors for wanting a conflation system, namely, having more than one set of maps to conflate, is only now being experienced by users outside of the Bureau of the Census. Digital map files were very expensive to create and, as a result, were very expensive to buy; and users generally did not have, nor did they seek, access to more than one digital file of the same region. However, with the proliferation of computer mapping services and applications and the subsequent reduction in digital map file costs, users can now avail themselves of multiple map representations of a single region, often produced by different sources for different applications. These same users now would like to be able to consolidate their diverse source materials into a single coherent map. At least one commercially available geographic information system has developed a conflation capability to allow the users to do exactly that [4].

Conflation System Development

Interactive computer graphics/visual methods. Alignment is the key to successful map feature matching and merging. If graphic images of the maps can be easily rubber-sheeted to align with each other, then verification of feature matches becomes a straightforward visual check. Rubber-sheeting, or, more formally, transformations that preserve topology, can be made to iron out local distortion, so that corresponding features of the two maps are not only recognizable, but they are actually coincident. Fast algorithms and fast drawing hardware permit rapid image rubber-sheeting. The original version of the conflation prototype system was screen-based and image-driven [5]; and it fully exploited the fast rubber-sheeting techniques. It allowed and even assisted the operator to select pairs of intersections to be matched. The process was interactive and iterative; with each additional selected matched pair, the operator would bring the image of one map into better alignment with the image of the other map. The image itself and a menu of tools for manipulating that image were the primary resources available to the operator [9].

One of the files had no auxiliary attribute information associated with the features [17]. Thus matching had to be based entirely on properties of the underlying line graphs of the maps. Matching based exclusively on line graph position and configuration was accomplished successfully through image manipulation to alignment followed by visual verification. Subsequent experiments at the Census Bureau [14] and practical applications elsewhere [4] have reconfirmed the original assessment of the efficacy of alignment as a visual tool for accomplishing conflation.

Batch processing/computational methods. Although alignment and visual verification do accomplish conflation, they do so at a considerable cost in time and effort. They prove that conflation is feasible as a computer-assisted manual process; and such a process may be totally acceptable to geographic information system users with two or three maps to conflate.

Indeed, conflation services offered to GIS users will likely remain highly interactive and image-driven [4].

However, for the task facing the Census Bureau of conflating 5,700 map pairs, an image-driven process requiring continuous operator intervention is expensive and also subject to considerable human error and inconsistency. In order to automate more fully the conflation system, a computerized approach to feature matching [8] was needed to replace the operator's skills at discerning feature similarities and differences between the map images. This goal was accomplished by first calculating a set of representative numerical values for each feature on each of the maps and then screening pairs of features for potential matches by comparing their sets of numerical values. Although the computerized comparison was inspired by visual considerations, the numerical assignments and checks were actually performed on the files themselves and not on a drawn image. The automated version, therefore, eliminated the need for drawing and re-drawing the maps on the screen at every stage of the iterative operation, and thereby shortened the time needed for the total processing.

CONFLATION SYSTEM STRUCTURE

System Outline

The Census Bureau's Conflation System [14] employs an iterative approach to map matching. That iterative approach may be sketched as an algorithm.

One begins by defining a sequence of test criteria to be applied successively to determine feature matches:

MATCH-CRITERIA(1), MATCH-CRITERIA(2),...,MATCH-CRITERIA(K),

The match criteria will compare numerical values associated with the map features to be matched.

The programs implementing these criteria receive, as input, a pair of maps, (possibly already partially matched), along with pre-computed or computable numerical values attached to feature lists. The programs return, as output, sets of pairs of matched features and lists of unmatchable features for each map. The Bureau of the Census Conflation System matching operation can be diagrammed succinctly as the following *K*-loop:

```
REPEAT FOR J = 1 TO K

    APPLY MATCH-CRITERIA(J)
      (RETURNS (MATCHES(J), UNMATCHABLES(J))
    ALIGN MAPS BASED ON MATCHES(1)...MATCHES(J)

END LOOP
```

The alignment of the maps is accomplished through triangulation and rubbersheeting procedures that will be explained below.

Feature Matching Routines and Algorithms

The match criteria *MATCH-CRITERIA(J)* need not all be different. Each rubber-sheeting operation will improve total alignment of the maps; and match criteria based on good alignment may fail in one iteration and succeed in the next. Some match criteria in use now are based on previous successful matches; and these match criteria will undoubtedly give different results when applied at different iteration points. One simple iteration rule is to apply a set of match criteria until they give no new matches before moving on to the next different set of match criteria.

All of the match criteria developed and tested at the Bureau of the Census contain a component based on nearness or proximity of features [8]. Nearness may refer to usual Euclidean distance or to pseudo-distances along a graph or network. Successively improved alignment through rubber-sheeting will progressively adjust Euclidean distances without changing topological pseudo-distances of the graph or network.

Feature matching initially refers to 0-cell matching, (or intersection matching, or point matching). Intersection matching is the easiest to define and the easiest to implement in a fashion that permits topologically consistent extension to the whole space. Matched points serve as vertices for triangulations of the space; and the triangulations permit each triangular region of one map to be associated with a corresponding region of the other map.

Matched 1-cells arise as induced matches when the corresponding end points of the 1-cells have been matched as 0-cell matches and when 1-cell shape is determined to match as well [13]. 2-cell matches are induced by matches along their boundaries, namely the 1-cells and 0-cells.

The 0-cells are matched using four primary tools:

(1) *Nearest Neighbor Pairings*. The 0-cells on one map are compared with the 0-cells of the other map after an initial alignment of the maps is made; and nearest neighbor pairs are identified. A pair of points is a nearest neighbor pair if the points are from different maps and each is the closest point from the other map to its paired point. Nearest neighbor pairs are found by applying a nearest point algorithm to create a sequence of points which alternates from one map to the other. Such a sequence stabilizes rapidly in a nearest neighbor pair. The nearest point algorithm uses a list ordering of the points called a Peano-key ordering which is based on interlacing the digits of a binary representation of the vertical and horizontal coordinates. The Peano-key list ordering is accessed through a B-tree because the list is updated throughout the conflation process.

Nearest neighbor pairs are candidates for matching if other match criteria tests are also met. Several of the other tests utilize the following two integer measures of local configuration:

(2) *Degree of a 0-cell*. The number of rays emanating from a 0-cell is called the degree or index of that 0-cell. Street intersections on the two maps are more likely to match if they are depicted with the same number of streets at the intersection. Similarly, the streets entering the intersection should be entering at approximately the same angles on the two different map depictions:

(3) *The Spider Function of a 0-cell*. The ray pattern at a 0-cell (that is, the emanating 1-cells) has infinitely many possibilities of rays and ray directions. In order to simplify the possibilities, the number of directions

was reduced to 16 and later 8 sectors. The eight sectors finally decided upon correspond to 45° pie slices in the principal directions of north, northeast, east, southeast, south, southwest, west, and northwest. The ray pattern is assumed to have at most one ray in each of the eight sectors (more than one ray in any sector will alter the spider function representation and reduce the chances for making a match).

The eight sectors in clockwise order are assigned consecutive bit positions (from right to left) in an 8-bit binary number, and the bit for a given sector is turned on if and only if there is a ray in that sector. The resulting number has been descriptively named the spider function of the 0-cell. With this function, an integer between 0 and 2^8-1 describes the ray pattern of the 0-cell. The binary number 01010101 (which is the decimal 85) represents the typical 4-street north-south-east-west intersection, for example. Intersection patterns which differ by a power of two are "close" in one of two geometric senses: either one pattern is missing a single street, but agrees everywhere else; or else one street is shifted, off by a single sector. By comparing the *degree* of a 0-cell as well as the spider function, the Bureau developed several simple measures of nearness of configuration.

(4) *Dependent Matching Routines*. Dependent matching routines are rules for applying relaxed matching criteria to 0-cells which are adjacent to already matched 0-cells. These routines use the 1-cell network of the map to precipitate additional matches. For that reason, it is important to store the topological relations of the map in order to facilitate network or graph traversals.

After each matching phase, matched points are moved into perfect alignment; and other points of the map are assigned to triangles whose vertices are the matched points. All of the unmatched points are then moved according to the movements of the vertices of the triangles which contain them by a joint triangulation [15] and rubbersheeting.

Triangulation Routines and Algorithms

For the conflation system developed at the Bureau of the Census, the rubber-sheeting transformations are defined on triangular subregions of each map; and the triangular subregions must cover each map and not overlap each other (except to share an edge). A polygonal region such as a map area may be subdivided into non-overlapping triangles. Such a subdivision into triangles is called a triangulation of the region. One type of triangulation, called the *Delaunay triangulation*, exhibits special properties [7] which make it particularly useful for the conflation system. The Delaunay triangulation is unique and locally updatable; and it may be built and updated in expected linear time with a suitable data structure. The following tools were developed to deal with triangulation requirements of conflation:

(1) *The Triangle-based Point Directory*. Since all 0-cells and other point features need to be transformed according to the triangle that contains them at any particular iteration, and since the triangles are changing with each iteration in a local fashion (in other words, not all of the triangles change), one may keep track of triangle containment by maintaining a triangle-based point directory and updating it with each triangulation update. A triangle-based point directory eliminates the need for determining the containing triangle of each point using some point-in-polygon test. As

the number of triangles increases, the rubber-sheeting routines do not require additional computer time to search for the containing triangle if the triangle-based point directory is used.

(2) *The Delaunay Triangulation Iterative Building Routines: Add-a-Vertex*. The Delaunay triangulation on a set of vertices may be built by adding one vertex at a time and re-triangulating after each addition. Each re-triangulation involves changing a few of the triangles in the neighborhood of the added point [10], and may be accomplished in constant expected time following a search to determine the neighborhood of the update. That search may be accomplished in constant time if the triangle-based point directory is used or in worst-case logarithmic time if a Peano-key/B-tree search strategy is employed.

(3) *Triangulation Extendability Results*. In order for the rubber-sheeting routines to be applied, the one-to-one correspondence of matched point pairs must be able to be extended to a one-to-one transformation of the entire triangulated spaces. Tests for extendability of the one-to-one transformations were developed [15]. When the tests fail, the vertex set must be modified, and ways to do this were developed as well [15].

(4) *The Delaunay Triangulation Modification Routines: Delete-a-Vertex*. The Delaunay triangulation may require modification in the form of vertex deletion due to non-extendability of the corresponding triangulation in the other space. A delete-a-vertex routine [12] provides a triangulation update based on re-triangulating the star-shaped polygon containing the vertex to be deleted.

After a map is triangulated and the image of the triangles is tested for one-to-one correspondence, a rubber-sheeting transformation is applied which moves every point of the map into a new position and aligns those point pairs which have been flagged as matches.

Rubber-Sheeting Routines and Algorithms

In their paper on rubber-sheeting, Marvin White and Pat Griffin [3] describe a *local affine transformation* of maps which sends triangles into triangles in such a way that shared edges of neighboring triangles are transformed in a consistent manner. The Census Bureau has adopted the rubber-sheeting transformation of White and Griffin, applied it to their own Delaunay triangulations, and expressed it in terms of simplicial coordinates for greater ease of manipulation and speed of computation.

The primary tool developed for rubber-sheeting is a computational algorithm for finding *simplicial coordinates* using linear equations for triangle edges [11]. The same computation serves to test for triangle containment of a point, since a point lies in a triangle if and only if its simplicial coordinates are between 0 and 1.

Local affine transformations preserve linearity and parallelism on each triangle; however, these continuous transformations are not differentiable at the edges of triangles. Because the rubber-sheeting transformations are applied only to discrete points (0-cells and shape points), however, there is no need to preserve differentiability everywhere. (Computer map lines are invariably made up of non-differentiable polygonal line segments anyway). After the 0-cells and shape points have been transformed, the connecting lines or 1-cells are redrawn to complete the map image.

Other non-affine rubber-sheeting methods have been considered and used by other implementers of conflation systems [4]. However, the other methods were not used at the Bureau of the Census because of their greater computational complexity and their potential topological inconsistencies.

CONFLATION SYSTEM PERFORMANCE AND EXPERIENCES

The application of the prototype system to real test files revealed considerable information about map differences and similarities. Moreover, these differences and similarities could be readily quantified and summarized [14]. The experiences at the Bureau of the Census dealt only with comparisons of USGS and Census Bureau files and should not be generalized. Nevertheless, others entering the field of conflation have confirmed some of the initial experiences.

Efficacy of proximity measures

For one particular pair of test maps of Fort Myers, FL, the efficacy of an alignment approach to matching can be illustrated by the following table showing the distribution of nearest neighbors by distance from each point on the Census map to its nearest neighbor on the USGS map when the neighbor is a matched point and also when the neighbor is not a match.

Frequency Distribution of Distance to the Nearest Neighbor

Distance Range (in meters)	Count of Nearest-Neighbor Matches within Range of Point	Count of Nearest-Neighbor Non-matches within Range of Point
0 to 5	162	-
5 to 10	359	-
10 to 15	272	4
15 to 20	132	8
20 to 25	70	14
25 to 30	19	25
30 to 40	13	54
40 to 50	3	90
50 to 60	2	227
60 to 70	-	302
70 to 80	1	134
80 to 100	-	86
100 to 200	1	82
200 to 400	-	8
400 and above	-	-

	Mean distance	Range	Std. Dev.
To Matching Nearest Neighbor	11.45	112.25	7.75
To Non-matching Nearest Neighbor	66.68	278.89	28.55

Table 1. Distance to Nearest Matches and Nonmatches After Initial Alignment

A good initial alignment can bring nearly all matchable pairs into proximity in such a way that the proximity relation almost becomes a necessary (but not sufficient) condition for matchability. Others have found that in some cases proximity may be also sufficient [4].

The average distance from any intersection to its nearest neighbor intersection on the same map is large compared to the average amount of distortion between maps; and this fact alone makes an iterative alignment approach very effective.

In general, two types of false classifications may occur. A map feature may be labelled incorrectly as having a match when indeed it does not (false positive); or a feature may be judged incorrectly not to match any feature when it has a true match (false negative). Because an iterative matching procedure identifies new matches at each stage, and, in general, does not flag non-matches as such until the final stage, missed matches may be corrected at a subsequent stage. False positive errors are less desirable and less managable than false negatives because at no point in the iteration procedure is there an automatic un-match capability for correcting false positives.

The table above shows that the distance between potential matches after initial alignment is an excellent tool for controlling both types of errors. Distance may be used for predicting both types of error and reducing one or the other: in the Fort Myers map, for example, if the threshold for matching is set at 20 meters, (that is, no matches are accepted unless the candidate pair are within 20 meters), then the measured probability of making a false negative error is 11%, and the probability of a false positive is 1%. By decreasing the threshold, false positives may be reduced further; however the increase in false negatives will require additional iterations of the file processing; and the threshold may even need to be relaxed in the final iterations in order to detect all matches.

The distance results above illustrate that nearest neighbor pairs are excellent candidates for matching if other match criteria tests are also met. The next section focuses on other tests which utilize the degree and the spider function, measures of local configuration:

Configuration measurement and findings

The representation of the spider function value as a hexadecimal integer has additional nice properties. It is a two-digit number; and each digit describes the street directional behavior in a four-sector band constituting a semi-circular region. A digit in the second position describes the same configuration as the same digit would in the first position except for a rotation of 180°.

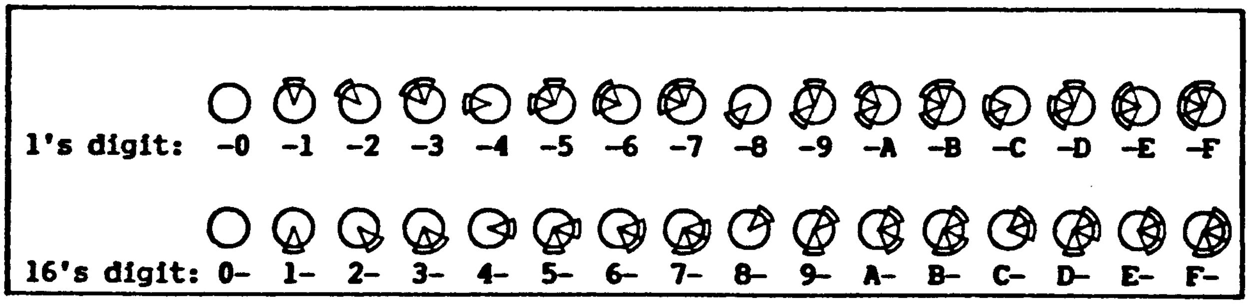

Figure 1. Hexadecimal and Sector Patterns for Spider Function.

A frequency distribution of spider function values for a map may be organized in a sixteen-by-sixteen table whose columns correspond to second (units) digit values and whose rows correspond to first (or sixteens) digit possibilities. In a highly urbanized area, for example, the frequency of the hexadecimal number 55, representing the north-east-south-west intersections, would be very large, and could help distinguish between urban and other areas. More generally, the frequency table establishes a kind of signature for the street network; and parts of the table, such as the diagonal, have special meaning. (The principal diagonal of the table consists of precisely those intersections all of whose streets continue straight through the intersection.) Two tables (one for the USGS map and one for the DIME map) showing the distribution of spider function values for all map intersections for a 25 square mile Fort Myers area are given below.

	-0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-A	-B	-C	-D	-E	-F
0-	0	36	7	2	38	29	2	-	5	2	1	1	5	6	1	-
1-	35	23	1	19	28	280	11	-	3	16	2	-	2	-	-	-
2-	9	3	4	6	1	9	10	-	3	4	9	1	1	-	1	-
3-	1	13	4	-	10	2	-	-	1	-	-	-	-	-	-	-
4-	28	22	4	14	22	315	12	-	3	18	7	-	6	4	-	-
5-	40	273	10	-	304	225	3	-	10	3	-	-	-	-	-	-
6-	6	10	4	1	4	2	4	-	-	-	5	-	1	-	-	-
7-	1	-	-	1	-	-	2	-	-	-	-	-	-	-	-	-
8-	8	-	1	2	3	13	2	-	3	10	21	-	3	-	-	-
9-	-	19	2	-	17	-	-	-	21	2	2	-	-	-	-	-
A-	2	2	10	-	10	-	2	-	29	3	9	-	-	-	-	-
B-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
C-	4	1	2	-	11	-	-	-	4	-	2	-	-	-	-	-
D-	5	1	1	-	2	-	-	-	1	-	1	-	-	-	-	-
E-	1	-	-	-	-	-	-	-	-	-	1	-	-	-	-	-
F-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Table 2. Spider function distribution for USGS intersections.

	-0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-A	-B	-C	-D	-E	-F
0-	-	93	22	-	80	16	-	-	21	1	1	-	-	6	-	-
1-	78	23	4	10	8	188	4	-	-	11	1	-	-	2	-	-
2-	14	2	5	4	4	12	7	-	1	1	6	2	3	-	-	1
3-	-	8	3	1	6	2	-	-	-	-	-	-	-	-	-	-
4-	77	10	5	9	31	204	13	1	5	21	5	1	14	1	-	-
5-	10	179	10	3	204	154	4	-	13	5	1	-	-	-	-	-
6-	3	6	3	-	10	7	3	-	2	-	5	-	1	-	-	-
7-	-	2	2	1	-	-	1	-	-	-	-	-	-	-	-	-
8-	12	1	2	1	3	8	2	-	2	1	17	-	9	-	-	-
9-	2	13	10	-	14	2	-	-	28	1	2	-	-	-	-	-
A-	2	5	8	-	8	1	-	-	25	3	12	-	4	-	-	-
B-	4	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-
C-	2	-	5	-	14	-	-	-	5	-	-	-	1	-	-	-
D-	7	-	-	-	3	-	-	-	-	-	-	-	-	-	-	-
E-	1	-	-	-	1	-	-	-	-	-	1	-	-	-	-	-
F-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Table 3. Spider function distribution for DIME intersections.

The above tables can orient an initial exploratory data analysis of intersection patterns of the area. One may display, for example, in the plane, all of those intersection points having a particular spider value (or a range of related spider values) and then proceed to apply pattern recognition techniques to the pattern [14].

The two preceding tables both exhibit unusual, but similar, distributions of their standard north-south-east-west (value 55) and their "T" (values 14, 41, 45, 54) intersections. The fact that there are more of every kind of "T" intersection than of the north-south-east-west type is due to frequent natural obstacles in the form of waterways that prevent streets from continuing through every intersection.

Since the patterns themselves are linked to spatial position, the tables shown above could further be decomposed according to subareas or subregions. Although the total number of entries would decrease, the fewer entries would then reflect more accurately local characteristics of the street network.

One may also display the spatial distribution of spider function values by simply plotting those values in their actual intersection locations.

Although condensing the network information at an intersection to a single number inevitably causes some loss of information, the resulting patterns lend themselves to many standard pattern recognition and analysis techniques. A number of standard references are available for such spatial analysis techniques. The pattern differences need to be viewed not only in terms of statistical error measurements, but also in terms of geometric relations of similarity shared by subsets of the spider function values [8].

CONCLUSIONS AND FUTURE DIRECTIONS

The evolution of conflation offers promising rewards for map users and analysts by making possible consolidations and comparisons that could not have been accomplished in the past. Moreover, the proliferation of digital map products has made users and geographic information systems designers aware of a need to combine those products meaningfully and systematically. Conflation at the Bureau of the Census will continue to develop as a tool of automation because of the volume of the task that inspired the system. Private developers can and should focus on image-based manual interactive systems because each user will have a small collection of maps to conflate and will want to accomplish the conflation task with continuous operator review during which the user's particular constraints may be enforced.

The conflation process will be used at the Bureau of the Census after the 1990 Census of Housing and Population to systematically update maps through identification and resolution of map differences. Measures of map quality will be established, based on number and magnitude of differences with base map sets. Considerable third-source review (aerial photographs or field testing) will be undertaken to resolve differences at that time.

Additional research is planned in the areas of joint triangulations and special map projections, statistical matching algorithms, and pattern recognition/image analysis techniques at the Bureau of the Census.

CONFLATION REFERENCES

1. Fagan, G. and H. Soehngen, 1987, "Improvement of GBF/DIME File Coordinates in a Geobased Information System by Various Transformation Methods and 'Rubbersheeting' Based on Triangulation," <u>Proceedings of 8th International Symposium on Automation in Cartography</u> (<u>AUTOCARTO 8</u>).

2. Gillman, D., 1985, "Triangulations for Rubber-Sheeting," AUTOCARTO 7.

3. Griffin, P., and M. White, 1985, "Piecewise Linear Rubber-sheet Map Transformations," The American Cartographer, v.12, no.2.

4. Lupien, A. and W. Moreland, 1987, "A General Approach to Map Conflation," AUTOCARTO 8 Proceedings.

5. Lynch, M. P., and A. Saalfeld, 1985, "Conflation: Automated Map Compilation—A Video Game Approach," AUTOCARTO 7 Proceedings.

6. Pavlidis, T., 1982, Algorithms for Graphics and Image Processing, Computer Science Press, Rockville, MD.

7. Preparata, F. and M. I. Shamos, 1985, Computational Geometry: An Introduction, Springer-Verlag, New York.

8. Rosen, B., and A. Saalfeld, 1985, "Matching Criteria for Automatic Alignment," AUTOCARTO 7 Proceedings.

9. Saalfeld, A., 1985, "Comparison and Consolidation of Digital Databases Using Interactive Computer Graphics," Census/SRD/Research Report Number 85-11, Washington, DC.

10. Saalfeld, A., 1985, "Direct Computation of Delaunay Triangulations," Internal Census Bureau Document.

11. Saalfeld, A., 1985, "A Fast Rubber-Sheeting Transformation Using Simplicial Coordinates," The American Cartographer, v.12, no.2.

12. Saalfeld, A., 1985, "Updating Delaunay Triangulations After Point Removal," Internal Census Bureau Document.

13. Saalfeld, A., 1986, "Shape Representation for Linear Features in Automated Cartography," v.1, Technical Papers of the 1986 ACSM-ASPRS Annual Convention.

14. Saalfeld, A., 1986, "Statistical Analysis of Map Differences," Proceedings of NCGA's GRAPHICS '86, also in ASA's Proceedings on Statistical Graphics, and to appear in The Journal of Official Statistics.

15. Saalfeld, A., 1987, "Joint Triangulations and Triangulation Maps," Proceedings of the ACM Third Annual Symposium on Computational Geometry, Waterloo, Ontario.

16. Serra, J., 1982, Image Analysis and Mathematical Morphology, Academic Press, New York, NY.

17. U. S. Geological Survey/Bureau of the Census, 1983, "Memo of Understanding for the Development of a National 1:100,000 Scale Digital Cartographic Data Base," Washington, DC.

18. White, M., 1981, "The Theory of Geographical Data Conflation," Internal Census Bureau draft document.