Optimized Feature Matching in Conflation

Linna Li [1], Michael F. Goodchild [1]

[1] Department of Geography, University of California, Santa Barbara, CA 93106, USA
{linna, good}@geog.ucsb.edu

1. Introduction

The rapid development of technologies makes it possible to collect vast volumes of geographical data. Examples of well-known datasets include US Census TIGER/Line files, USGS datasets, and DigitalGlobe. Meanwhile, many applications of GIS require data from more than one source, and these data may be initially created for various purposes, in different formats, and at various scales. For example, in disasters such as Hurricane Katrina, effective management of catastrophic events requires coordination of a wide range of geographical data such as DEM, land use, facility locations, *etc*. In other applications, update of an existing database is required when a change database is available.

Conflation is the process of combining information from two or more related datasets and acquiring knowledge that cannot be obtained from any single data source alone. The difficulty of this process depends on the complexity of representation and the size and accuracy of the involved datasets. There are generally three phases in geographical data conflation: feature matching, position transformation, and attribute merging. Feature matching involves the identification of features in multiple datasets that represent the same entity in reality, which is the focus of this paper.

Feature matching in conflation is usually performed according to the similarity between potential feature pairs. If two features are represented similarly in terms of locations, shapes, and relationships with surrounding features, it is probable that they represent the same entity in the real world. Samal et al. (2004) summarized a set of similarity measures, including categorical similarity, string similarity, and shape similarity. Similarity measures commonly used in feature matching can be classified into three types: geometry, attribute, and topology (Devogele 2002, Saalfeld 1988, Walter and Fritsch 1999), and many useful similarity metrics have been developed to compare the resemblance between features (Arkin et al 1991, Bel Hadj Ali 1997, Harvey and Vauglin 1997, Lemari and Raynal 1996, Vauglin and Bel Hadj Ali 1998).

Although the criteria for feature matching vary in different applications, a common strategy in previous work is the sequential workflow of matching: pairs of matched features are identified one after another. An obvious issue with this method is that when a feature is matched to a wrong counterpart in the other dataset, no remedy can be made to correct this error. A main purpose of this paper is to propose an alternative matching strategy to solve this problem.

2. Methodology

In automatic feature matching, we need to develop an objective function whose value is an effective indicator of global goodness for making matches. A natural choice would be to minimize the total dissimilarity between corresponding features. However, as discussed above, existing feature matching methods in the literature generally adopt a sequential greedy strategy that fails to achieve a global optimum (*e.g*. Cobb et al.

1998, Filin and Doytsher 2000). This section proposes feature matching as an assignment problem that could be solved by an optimization model.

2.1 Greedy vs. optimization methods

The essential characteristic of a greedy method is that it makes choices one after another in a series of steps, and at each step selects the option that makes the greatest improvement to the objective function. When a new candidate is added to the solution set, previous choices cannot be changed and as a result the eventual solution may be non-optimal. In feature matching, if a feature is incorrectly matched to another feature in a previous step, this error will not be rectified in later steps.

To overcome the shortsightedness of greedy methods, we alternatively identify all pairs of matched features simultaneously using an optimization method. The goal of this model is to find the global optimal solution from all possible choices, by minimizing an objective function that is subject to a set of constraints. Thus, the feature matching problem is formulated mathematically as follows. The objective function is

$$Minimize \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij} z_{ij} \tag{1}$$

where *i,j* are indices for features in the first and second dataset respectively, *n* is the number of features in each dataset, and c_{ij} is the dissimilarity between feature *i* and feature *j*. The variable z_{ij} is a match indicator between two features, taking value 1 if a match is made and 0 otherwise, *i.e.*

$$z_{ij} = \begin{cases} 1, \textit{if a match is made between features i and j} \\ 0, \textit{otherwise} \end{cases} \tag{2}$$

subject to

$$\sum_{j=1}^{n} z_{ij} = 1, \quad \forall i \tag{3}$$

$$\sum_{i=1}^{n} z_{ij} = 1, \quad \forall j \tag{4}$$

The first constraint (Equation 3) requires that every feature in the first dataset is matched to exactly one feature in the second dataset, and the second constraint (Equation 4) requires that every feature in the second dataset is matched to exactly one feature in the first dataset.

2.2 Feature matching as an assignment problem

The above optimization model is well known in the operations research and computer science as the assignment problem (Hillier and Lieberman 2004). Since the goal of feature matching is to identify corresponding features that represent the same entity in reality, the task of this assignment problem is to assign each feature in one dataset to its counterpart in the other dataset, with the objective of minimizing the total dissimilarity between all matched pairs. The assignment problem is known to be a

polynomial time problem, which means that the run time for solving this problem is no greater than a polynomial function of the problem size: the number of features in the datasets in this case. This can be achieved either with specialized algorithms such as the Hungarian algorithm (Kuhn 1955) or by formulating it as a Linear Programming (LP) problem and solving it using a standard LP package.

3. Experiments

We apply two kinds of greedy methods and the optimization method to both hypothetical point datasets and real street network datasets. Greedy1 is a regular greedy method, and Greedy2 adds a random component in order to jump out of local optima by multiple runs. For details of these two greedy methods and methods for generating hypothetical data, see the work by Li and Goodchild (2010). The number of point features in each pair of datasets range from 10 to 100 with an interval of 5. The dissimilarity criterion for feature matching is the Euclidean distance between points. As the density of features increases, the percentage of correct matches by the two greedy methods decreases drastically, while the percentage of correct matches by the optimization method is relatively stable, close to 100% (Figure 1).

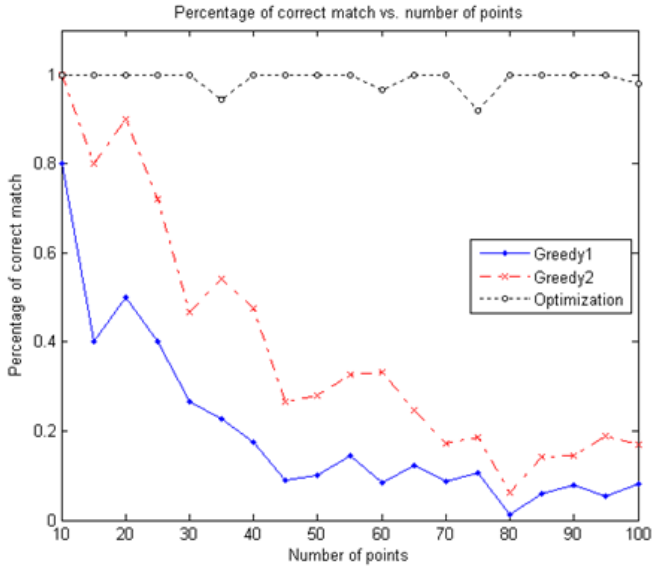
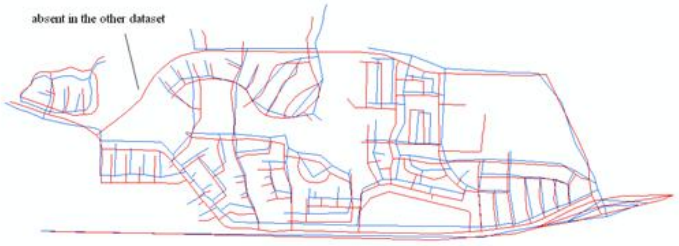


Figure 1. Percentage of correct matches vs. number of points.

The criterion for feature matching is the Hausdorff distance between polylines since it characterizes the proximity of two linear features particularly well (Abbas 1994), and the features to be matched are individual streets. Two versions of the same street network of a neighborhood in Goleta are displayed in Figure 2(a), and two versions of streets in downtown Santa Barbara are displayed in Figure 2(b). These datasets are more complex than the previous hypothetical point datasets, with different numbers of polylines and different numbers of vertices composing polylines. There are more than 200 features in Datasets (a) and more than 1000 features in Datasets (b). As demonstrated in Table 1, the total distance between matched pairs using the optimization method is smaller than that using the greedy methods. Furthermore, the mismatch rate is significantly reduced using the optimization method in both cases. Computation time for different methods is presented in Table 2. The running time for all three methods increases as the data size grows larger, with optimization increasing slightly faster than greedy methods. To explore possible solutions, we divide the larger

datasets with over 1000 features into two subareas and solve each part separately. As a result, the total computation time for solving the same datasets using the optimization method decreases from 805.6s to 145.8s as the good matching result is maintained. This indicates that a divide-and-conquer algorithm may be employed to take advantage of the positional information of geographic data and speed up the matching process.



(a)

(b)

Figure 2. Street networks in Santa Barbara, CA.

Table 1. Results of feature matching for street network datasets.

	(a)		(b)	
	Total distance	Mismatch rate	Total distance	Mismatch rate
Greedy1	13104	11.86%	42850	2.86%
Greedy2	13078	11.44%	45572	4.09%
Optimization	12369	2.97%	42454	1.72%

Table 2. Computation time for different methods (seconds).

	(a)	(b)
Greedy1	2.6	310.2
Greedy2	0.7	18.5
Optimization	3.3	805.6

4. Conclusion

Feature matching is one of the crucial components in conflation. To the best of our knowledge all previously published feature matching methods adopt a greedy strategy, which may lead to frequent mismatch errors. The nature of the greedy method makes it difficult to rectify errors in previous steps once the match is made. This paper proposes a new strategy for automatic feature matching in conflation: the matching process is regarded as an assignment problem that takes into account all potentially matched pairs simultaneously by minimizing the total distance of all pairs in a similarity space. The matching results in the experiments demonstrate that the optimization strategy achieves a lower mismatch rate given the same criteria.

There are still some interesting research questions in geographical feature matching using the optimization method. Our ongoing work is investigation of 1:*m*, *m*:1, and *m*:*n* correspondence in feature matching. In addition, the efficiency of the algorithm may degrade with increasing problem size, so we plan to investigate general performance-improving strategies from computational geometry such as divide-and-conquer (Preparata and Shamos 1985) and parallel computing. As shown in Section 3, partitioning of a large dataset significantly reduces computation time.

Acknowledgements

This research is funded by the National Geospatial-Intelligence Agency through the NGA University Research Initiative Program (NGA-NURI grant HM1582-10-1-0007).

References

Abbas, I., 1994, Base de donnes vectorielles et erreur cartographique: problmes poss par le contrle ponctuel; une mthode alternative fonde sur la distance de Hausdorf. Computer Science. Paris, Universit de Paris VII.

Arkin, E.M., Chew, L.P., Huttenlocher, D.P., Kedem, K. and Mitchell, J.S.B., 1991, An efficiently computable metric for comparing polygonal shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(3): 209-216.

Bel Hadj Ali, A., 1997, Appariement geometrique des objets gographiques et tude des indicateurs de qualit. Saint-Mand (Paris), Laboratoire COGIT.

Cobb, M. A., Chung, M. J., Foley III, H., Petry, F.E. and Shaw, K.B., 1998, A rule-based approach for the conflation of attributed vector data. *Geoinformatica*, 2(1):7-35.

Devogele, T., 2002, A new merging process for data integration based on the discrete Frechet distance. In: D. Richardson and P. van Oosterom (eds), *Advances in Spatial Data Handling: 10th International Symposium on Spatial Data Handling*, New York, Springer Verlag: 167-181.

Filin, S. and Doytsher, Y., 2000, The detection of corresponding objects in a linear-based map conflation. *Surveying and Land Information Systems*, 60(2):117-128.

Harvey, F. and Vauglin, F., 1997, No Fuzzy Creep! A clustering algorithm for controlling arbitrary node movement. *AutoCarto 13*, Seattle, ASPRS/ASCM.

Hillier, F. S. and Lieberman, G. J., 2004, *Introduction to Operations Research* (McGraw-Hill).

Kuhn, H.W., 1955, The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2, 83-97.

Lemari, C. and Raynal, L., 1996, Geographic data matching: First investigations for a generic tool. GIS/LIS '96, Denver, Co, ASPRS/AAG/URISA/AM-FM.

Li, L. and Goodchild, M.F., 2010, Automatically and accurately matching objects in geospatial datasets. *Theory, Data Handling and Modelling in GeoSpatial Information Science*. (Hong Kong, 26-28 May, 2010).

Preparata, F. P. and Shamos, M. I., 1985, *Computational Geometry: An Introduction* (New York, NY: Springer-Verlag New York, Inc.).

Saalfeld, A., 1988, Conflation automated map compilation. *International Journal of Geographical Information Systems*, 2(3): 217-228.

Samal, A., Seth, S. and Cueto, K., 2004, A feature-based approach to conflation of geospatial sources. *International Journal of Geographical Information Science*, 18(5):459-489.

Vauglin, F. and Bel Hadj Ali, A., 1998, Geometric matching of polygonal surfaces in GISs. ASPRS Annual Meeting, Tampa, FL, ASPRS.

Walter, V. and Fritsch, D., 1999, Matching spatial data sets: a statistical approach. *International Journal of Geographical Information Science*, 13(5): 445-473.