CSIS Discussion Paper No. 34

A Computational Procedure for Making Seamless Map Sheets

Takashi Sato*, Yukio Sadahiro**, and Atsuyuki Okabe**

JANUARY, 2001

*Department of Urban Engineering, University of Tokyo
**Center for Spatial Information Science, University of Tokyo

Center for Spatial Information Science
University of Tokyo
7-3-1, Bunkyo-ku, Tokyo 113-8656, Japan

January 11, 2001

A Computational Procedure for Making Seamless Map Sheets (seamless map sheets)

Takashi Sato*, Yukio Sadahiro**, and Atsuyuki Okabe**

*Department of Urban Engineering, University of Tokyo
**Center for Spatial Information Science, University of Tokyo

Contact: Yukio Sadahiro
Center for Spatial Information Science and
Department of Urban Engineering, University of Tokyo
7-3-1, Hongo, Bunkyo-ku, Tokyo 113-8656, Japan
Phone: +81-3-5841-6273
Fax: +81-3-5841-8521
E-mail: sada@okabe.t.u-tokyo.ac.jp

Keywords: seamless map sheets, GIS, spatial database, transformation, computational procedure

Acknowledgment: This paper is an extended version of Sato et al. (2000) which has been published in Japanese. Authors express thanks to those who gave us valuable comments on the earlier version of the paper.

A Computational Procedure for Making Seamless Map Sheets

Abstract

Map sheets are often used as basic spatial units for managing spatial data produced from paper maps. This often results in incompatibility between adjacent map sheets - spatial objects do not cross the boundaries smoothly and even the boundaries themselves do not match their neighbors exactly. To solve the problem this paper proposes a computational procedure for making seamless map sheets. Line objects digitized separately in different map sheets are considered, which are frequently used to represent road networks, gas pipelines, and boundaries of polygon objects. The procedure consists of three steps: 1) extraction of end nodes, 2) detection of matching nodes, and 3) transformation of the map sheet. Each step goes interactively so that unexpected errors can be avoided by human observation. To test the validity of the procedure, map sheets are combined containing the road network data of Tokyo 23-ku area, Japan.

1 Introduction

Paper maps are usually compiled as separate map sheets. In general, rectangular lattices are used as basic units of map sheets; in Japan, for example, 1/25,000 topographic maps are provided as 1.5 * 2.5 km rectangular maps. Landsat TM images use 180 * 180 km square lattices.

This tradition still exists in spatial database - the digital form of paper maps. Map sheets are often used as basic spatial units for managing spatial data produced from paper maps. Maps are digitized separately and compiled as a set of map sheets in GIS. However, it is obviously desirable for GIS users to use sheetless and seamless spatial data because they wish to study regions larger than one map sheet. This complaint applies to not only local analysis of geographical phenomena such as urban landuse and traffic patterns but also sciences on a global scale, say, earth environment science and global climatology.

Unfortunately, seamless spatial data cannot be easily obtained when paper maps are separately digitized (Beard and Chrisman 1988). In theory, spatial data can be seamless if individual map sheets are correctly digitized - if their reference to the earth's surface and the location of spatial objects are correctly recorded in GIS. In practice, however, various types of errors creep in map digitizing process such as displacement of control points, distortion of papers, and human errors in manual operations. These errors cause disagreement among map sheets: spatial objects do not cross the boundaries smoothly and even the boundaries themselves do not match their neighbors exactly.

There are two approaches to obtain seamless data from separate map sheets: global and local transformations. In either case one or several map sheets are transformed to meet the adjacent ones. The basic difference between them lies in the scale of transformation. Global methods transform the whole area of maps so that all the spatial objects in map sheets may be affected. Local methods, on the other hand, work only around map boundaries, preserving spatial objects near the center of map sheets.

In global methods, transformation is usually limited to the affine transformation to avoid unnatural deformation of maps. Because of this global methods inevitably fail to match all the end nodes with those in adjacent map sheets (Petersohn and Venderohe 1982; White and Griffin 1985). This implies that sliver polygons still remain after the transformation if applied to polygon data. Local expansion and shrinkage of paper maps cannot be retrieved appropriately.

Among local methods, the nearest matching method is most frequently used in GIS. This method moves end nodes to their nearest ones so that lines are exactly connected between map sheets. A fuzzy tolerance is used to avoid excessive displacement of nodes. Since the nearest matching method is easy to understand and its implementation is

straightforward, it is popular in geography and cartography; some systems can automatically merge map sheets by the nearest matching method.

An extension of the nearest matching method is an algorithm called ZIPPER proposed by Beard and Chrisman (1988). It uses two parameters to detect matching nodes and to control the distortion area of maps. Consequently, it smoothly transforms the original data to produce seamless maps containing no sliver polygons.

Unlike global methods, local methods cannot appropriately retrieve global distortion of maps which can be caused by systematic errors in digitizing process and numerical calculation. Therefore, as Beard and Chrisman (1988) noted, a combination of global and local methods seems a desirable option because it can resolve both global and local errors. Following the line of this hybrid approach, this paper proposes a new computational procedure for making seamless map sheets. The procedure consists of three steps: 1) extraction of end nodes, 2) detection of matching nodes, and 3) transformation of the map sheet. Each step goes interactively so that unexpected errors can be avoided by human observation.

In the following we consider line objects without attribute information including identification numbers; this type of line data are frequently used to represent road networks, gas pipelines and boundaries of polygon objects. We assume that spatial data are managed in rectangular lattice, and that one of the map sheets is distorted so that it has to be transformed to match its surrounding sheets (Figure 1). The procedure, however, is also applicable to matching of two adjacent sheets. In the following sections, each step of the procedure is described successively. In Section 5, a set of road network maps are combined to test the validity of the procedure. The last section summarizes the conclusions with discussion.

Figure 1 A distorted map sheet (thick lines) that has to be transformed to match its surrounding sheets.

2 Extraction of end nodes

Suppose a rectangular lattice used for managing a set of map sheets containing line objects. Let R_0 be a rectangular region of the lattice whose corresponding map sheet is distorted. We call the region R_0 the *standard unit*. Our objective is to transform the map sheet to make it compatible with its surrounding sheets. Note, however, that this does not mean that the surrounding sheets are perfectly correct but that they are relatively correct compared with the target sheet.

The region covered by the distorted map sheet is called the *distorted unit* denoted by R'. The distorted unit may not be rectangular because map distortion does not always

preserve the original shape of maps. The region that originally has to be covered by the distorted map is called the *true unit* which is denoted by R. The true unit is not always identical with R_0 because errors usually exist in the surrounding sheets. Figure 2 shows an example of the three units R_0, R, and R'. The process of merging map sheets can be interpreted as to find a one-to-one mapping from R' to R taking R_0 into consideration.

Figure 2 An example of the standard unit R_0 (gray-shaded area), true unit R (solid lines), and distorted unit R' (broken lines).

Boundaries of R_0, R and R' are called the *standard*, *true*, and *distorted boundaries*, and denoted by ∂R_0, ∂R, and $\partial R'$, respectively. In general, the standard boundary ∂R_0 is explicitly recorded in spatial database or at least computable from known coordinates. The true and distorted boundaries, on the other hand, are not recorded so that we have to estimate them from the data.

Line objects are distributed in the distorted unit R'. The number of lines linked to a node is called the *order* of the node. Nodes are classified by their order, and the set of nodes of order i in R' is denoted by Ω_i. The set Ω_1 represents the set of end nodes. Let $\#(\Omega_i)$ be the number of nodes in Ω_i, and Ω be the set of all the nodes in R', that is, $\bigcup_i \Omega_i$.

The first step to merge map sheets is to extract end nodes, candidates for nodes to be matched with those in adjacent map sheets. They are classified into two types: *boundary nodes* and *interior nodes*. The former is a set of nodes that are expected to be exactly on the distorted boundary $\partial R'$. They usually have their corresponding nodes in the adjacent map sheets and consequently quite important in merging map sheets. The latter is also a set of nodes that seem to match nodes in the adjacent sheets though located inside $\partial R'$.

Extraction of boundary and interior nodes is a computational iterative process: we first set the initial candidates and update them iteratively until a predetermined condition is satisfied. The initial candidates are extracted from Ω_1 as follows. Since boundary and interior nodes are usually concentrated in the "neighborhood" of the standard boundary ∂R_0, the ratio of $\#(\Omega_1)$ to $\#(\Omega)$ is extremely high near ∂R_0. Mathematically, the ratio is represented by

$$r(x) = \frac{\#\left(\Omega_1 \in B(x)\right)}{\#\left(\Omega \in B(x)\right)} - \frac{\#\left(\Omega_1 \in \left(R' \setminus B(x)\right)\right)}{\#\left(\Omega \in \left(R' \setminus B(x)\right)\right)}, \tag{1}$$

where $B(x)$ is the inner buffer region of distance x from ∂R_0. The value of x maximizing Equation (1) is denoted by $x_{\max}$, and the inner buffer region $B(x_{\max})$ is considered to be

the "neighborhood" of ∂R_0. End nodes in $B(x_{\max})$ and outside of R_0 are the initial candidates for boundary nodes, while those in $B(2x_{\max}) \setminus B(x_{\max})$ are candidates for interior nodes.

Figure 3 Initial candidates for boundary (white circles) and interior nodes (double circles). Gray-shaded area represents the inner buffer region $B(x_{\max})$.

Initial candidates for boundary and interior nodes usually contain inappropriate nodes, that is, nodes that do not have corresponding nodes in the adjacent map sheets. To eliminate them we iteratively estimate the distorted boundary $\partial R'$ from candidate nodes and update them by dropping the farthest one from the estimated boundary $\partial \hat{R}'$ until all the candidates for boundary nodes are within the distance $x_{\max}/2$ from $\partial \hat{R}'$. The distorted boundary is estimated by fitting the line that minimizes the sum of the square distance to the candidates for boundary nodes. Figure 4 illustrates an example of this process.

Figure 4 Update of candidates for boundary nodes. Broken lines represent the estimated boundary $\partial \hat{R}'$. Black circles indicate boundary nodes rejected during the update process.

3 Detection of matching nodes

Having extracted end nodes in R', we have to detect their corresponding nodes in the adjacent map sheets. In theory, all of order 1 nodes in the adjacent sheets are candidates for corresponding nodes. However, it is practically sufficient to examine only the nodes in the "neighborhood" of ∂R_0 which is given by the buffer operation.

Among the end nodes in R' we first examine only the boundary nodes. We assign them to the nearest nodes in the adjacent map sheets, and adopt node pairs that are the nearest neighbors with each other (Figure 5a). To improve the accuracy of matching, we then drop the p percent (say, five percent) of the pairs whose node-to-node distance is relatively long.

We next investigate the interior nodes and the boundary nodes rejected in the previous step together. One can also include the nodes rejected in the update process of candidates for boundary nodes, if necessary. The procedure is quite similar; we assign the nodes to the nearest nodes in the adjacent map sheets, and adopt node pairs that are the nearest neighbors with each other (Figure 5b). We again drop the p percent of the pairs whose node-to-node distance is relatively long. The node pairs detected in these two steps are considered to be "matching" (see Sato 2001 for details).

Detection of matching nodes plays a key role in merging map sheets. Consequently, it is desirable for GIS users to examine the result of detection visually and revise it if

necessary. The system developed by Sato (2001) provides the function of visualizing matching nodes and their displacement vectors.

Figure 5 Detection of matching nodes. Solid lines represent line objects in the distorted map sheet while broken lines in its adjacent sheet. White and black circles indicate the boundary and interior nodes, respectively. (a) Investigation of the boundary nodes. Node pairs encircled by broken ellipses are the nearest neighbors with each other. (b) Investigation of the interior nodes and the boundary nodes rejected at the previous step.

Using the matching nodes detected, we can estimate the true and distorted units R and R'. Let N_i' be the ith node in the distorted map sheet detected in the above procedure (i=1, ..., n; N_{n+1}'=N_1'). We assume that the nodes are numbered counterclockwise with respect to the gravity center of the standard unit R_0 denoted by G. The node in the adjacent sheets corresponding to N_i' is denoted by N_i. The polygons $N_1N_2N_3...N_n$ and N_1'N_2'N_3'...N_n' are good estimators of R and R', respectively.

4 Transformation of the map sheet

Having detected matching nodes, we finally transform the distorted map sheet to obtain seamless spatial data. Transformation of the map sheet proceeds by two steps: a global transformation and a set of local transformations.

In general, local transformations are not desirable because they often yield unnatural distortion at the boundary of regions where different transformations are applied, even if the transformation functions are smoothly continuous across the regions. On the other hand, seamless map sheets cannot be obtained only by the global transformation. We thus adopt a hybrid approach of these two methods in order to obtain seamless map sheets without unnatural local distortions.

In the global transformation, nodes in the distorted map sheet are moved by the affine transformation that minimizes the sum of the square distance between matching nodes. The affine transformation includes not only the similarity transformation but also the mirroring and shearing transformations. If the mirroring and shearing transformations are to be avoided, the similarity transformation should be chosen which permits only the translation, rotation, and scaling of spatial objects. The transformation can also be limited only to the translation and rotation to keep the map scale.

The global transformation is followed by a set of local transformations. The distorted unit R' is divided into a set of triangular regions each of which is given by G, N_i', and N_{i+1}' (Figure 6, broken lines). Nodes are added to the line objects that cross the boundaries of triangles. The local transformation function is determined for each

triangular region - transformation from the triangle $\Delta GN_i'N_{i+1}'$ into the trilateral ΔGN_iN_{i+1} (Figure 6, solid lines). Note that the resulted regions may not be triangles but may contain curved edges; it depends on the transformation functions.

Figure 6 Tessellations of *R* (solid lines) and *R'* (broken lines).

Let us consider the transformation from $\Delta GN_i'N_{i+1}'$ into ΔGN_iN_{i+1}. Set the origin of the coordinate system at *G*. The locational vector of a point in $\Delta GN_i'N_{i+1}'$ is represented by two parameters *s* and *t*:

$$\mathbf{x}' = s\{t\mathbf{x}_i' + (1-t)\mathbf{x}_{i+1}'\} \quad (0 \le s,t \le 1), \tag{2}$$

where $\mathbf{x}_i'$ is the locational vector of N_i'. Let $T_{i,i+1}(\mathbf{x}')$ be the transformation function at $\mathbf{x}'$, that is, the point at $\mathbf{x}'$ in $\Delta GN_i'N_{i+1}'$ is moved to $T_{i,i+1}(\mathbf{x}')$. The displacement function $\Delta T_{i,i+1}(\mathbf{x}')$ is then given by

$$\Delta T_{i,i+1}(\mathbf{x}') = T_{i,i+1}(\mathbf{x}') - \mathbf{x}'. \tag{3}$$

We adopt the transformation whose displacement function is

$$\Delta T_{i,i+1}(\mathbf{x}') = \gamma(s)\{t\Delta T_{i,i+1}(\mathbf{x}_i') + (1-t)\Delta T_{i,i+1}(\mathbf{x}_{i+1}')\}, \tag{4}$$

where *s* and *t* are given by Equation (2). The function $\gamma(s)$ is a monotonically increasing function of *s* that satisfies

$$\gamma(0) = 0, \tag{5}$$

and

$$\gamma(1) = 1. \tag{6}$$

Substituting Equations (3) into (4), we obtain the transformation function

$$\begin{aligned} T_{i,i+1}(\mathbf{x}') &= s\{t\mathbf{x}_i' + (1-t)\mathbf{x}_{i+1}'\} + \gamma(s)\{t(\mathbf{x}_i - \mathbf{x}_i') + (1-t)(\mathbf{x}_{i+1} - \mathbf{x}_{i+1}')\} \\ &= t\{\gamma(s)\mathbf{x}_i + (s-\gamma(s))\mathbf{x}_i'\} + (1-t)\{\gamma(s)\mathbf{x}_{i+1} + (s-\gamma(s))\mathbf{x}_{i+1}'\} \end{aligned}. \tag{7}$$

The transformation given by Equation (7) keeps the topology of line objects in *R'*; the gravity center *G* is not moved while the vertices N_i' and N_{i+1}' are exactly moved to N_i and N_{i+1}, respectively; line segments parallel to $N_i'N_{i+1}'$ are not curved after transformation. To keep the topology of spatial objects is critical in GIS because it represents the structure of spatial objects. The monotonicity of $\gamma(s)$ assures that the displacement of spatial objects increases monotonously with *s*; points near the gravity center *G* move slightly while those around the edge $N_i'N_{i+1}'$ move greatly. Figure 7 shows an example of the transformation.

Figure 7 Transformation of $\Delta GN_1'N_2'$ (broken lines) into ΔGN_1N_2 (solid lines).

There are a wide variety of functions satisfying the conditions mentioned above. In this paper we use the power function

$$\gamma(s) = s^{\alpha}. \tag{8}$$

The parameter α determines the range of transformation; a small value of α indicates that the transformation works globally; as the value of α increases the transformation becomes local, that is, only the spatial objects near the boundaries are moved. The trilateral ΔGN_iN_{i+1} becomes a triangle when α=1.

The parameter α is so determined as to maximize the correlation coefficient of the bidimensional regression where the points in $N_1'N_2'N_3'...N_n'$ and $N_1N_2N_3...N_n$ are the dependent and independent variables, respectively (for details of the bidimensional regression, see Tobler 1994 and Nakaya 1997). The correlation coefficient ρ is given by

$$\rho = 1 - \frac{\sum_{i=1}^{n} \int_{\mathbf{x}' \in GN_i' N_{i+1}'} \left| \mathbf{x}' - T_{i,i+1}(\mathbf{x}') \right|^2 \mathrm{d}\mathbf{x}'}{\sum_{i=1}^{n} \int_{\mathbf{x}' \in GN_i' N_{i+1}'} \left| \mathbf{x}' - \bar{\mathbf{x}}_{i,i+1}' \right|^2 \mathrm{d}\mathbf{x}'}, \tag{9}$$

where $\bar{\mathbf{x}}_{i,i+1}'$ is defined by

$$\bar{\mathbf{x}}_{i,i+1}' = \frac{\int_{\mathbf{x}' \in GN_i' N_{i+1}'} \mathbf{x}' \, \mathrm{d}\mathbf{x}'}{\int_{\mathbf{x}' \in GN_i' N_{i+1}'} \mathrm{d}\mathbf{x}'}. \tag{10}$$

The correlation coefficient ρ represents how close the transformation from $N_1'N_2'N_3'...N_n'$ to $N_1N_2N_3...N_n$ is to the similarity transformation. Consequently, a large value of ρ is desirable because the similarity transformation does not yield unnnatural distortion of spatial objects. Integral terms in Equations (9) and (10) are practically calculated by numerical integration.

Local transformation functions are calculated for individual triangular regions, and applied to all the nodes of line objects including those added in the triangulation of *R'*. This, however, often changes straight lines running across triangles into polygonal lines because the transformation function is different among the regions. We thus finally remove the redundant nodes on the boundaries of triangles to restore straight lines. The topology of line objects is examined at each step of node removal to preserve the original topology. Figure 8 shows an example of the transformation process.

Figure 8 Transformation of line objects. (a) Original (distorted) lines and triangles, (b) triangulation (nodes are added), (c) transformation, (d) removal of redundant nodes.

5 Empirical study

To test the validity of the procedure proposed above, we combine map sheets containing the road network data of Tokyo 23-ku area, Japan. About 250 map sheets cover the Tokyo 23-ku area, each of which is a rectangle of 1.5 * 2.5 km.

We have transformed three map sheets that do not match their surrounding sheets,

but show only the result of one of the sheets S_0 due to the limitation of space. Figures 9 and 10 shows the data in the original sheet and some mismatches between S_0 and its adjacent map sheet.

Figure 9 Original data of the map sheet. Solid and broken lines indicate the road network in the S_0 and its adjacent sheets, respectively.

Figure 10 Mismatches between S_0 and its adjacent map sheet.

To transform S_0 we first extracted its end nodes, candidates for nodes to be matched with those in adjacent map sheets, using the method proposed in Section 2. The distance $x_{\max}$ which determines the "neighborhood" of ∂R_0 was set to 0.240m, which yielded 110 initial candidates for boundary nodes and 26 interior nodes. The initial candidates for boundary nodes were then updated by the iterative estimation of the distorted boundary, and as a result, 83 boundary nodes were extracted. Figure 11 shows the boundary and interior nodes in S_0.

Figure 11 Boundary and interior nodes extracted in S_0. White and black circles indicate the boundary and interior nodes, respectively.

Though the distance $x_{\max}$ was considerably small, extraction of end nodes seemed successful by visual observation. We thus accepted the results and proceeded to detect their corresponding nodes in the adjacent map sheets. At the first step of detection, only the boundary nodes in S_0 were examined. Among 83 boundary nodes, 67 nodes were assigned to their corresponding nodes in the adjacent sheets. We then dropped the five percent of the pairs whose node-to-node distance is relatively long. At the second step, we investigated the interior nodes and the boundary nodes rejected together, and found corresponding nodes for 33 interior and boundary nodes among 42. We again dropped the five percent of the pairs whose node-to-node distance is relatively long, and obtained 100 matching node pairs. Figure 12 shows several examples of the nodes that failed to find their corresponding nodes.

Figure 12 Nodes that failed to find their corresponding nodes (black circles). Solid and broken lines indicate the road network in the S_0 and its adjacent sheets, respectively.

Using the matching nodes detected we finally transformed the map sheet S_0 by global and local transformations. Global transformation is so determined as to minimize

the sum of the square distance between matching nodes. After the global transformation, the map sheet S_0 is divided into 100 triangular regions as shown in Figure 13 in each of which a local transformation is applied.

Figure 13 Tessellations of the map sheet S_0.

The parameter α in the function $\gamma(s)$ which determines the range of transformation is set to 0.280 that maximizes the correlation coefficient of the bidimensional regression mentioned earlier. Since the value is relatively small, the transformation is performed locally around the boundaries. Local transformations were followed by removal of redundant nodes on the boundaries of triangles to restore straight lines. Figure 14 shows several examples of the removal of redundant nodes, which effectively demonstrates the usefulness of this operation. Figure 15 shows the final result of transformation in which the road network is exactly compatible between adjacent map sheets.

Figure 14 Examples of the removal of redundant nodes. Road networks before (broken lines) and after (solid lines) the removal.

Figure 15 Final production of the transformation.

6 Concluding discussion

In this paper we have proposed a computational procedure for making seamless map sheets. The procedure consists of three steps: 1) extraction of end nodes, 2) detection of matching nodes, and 3) transformation of the map sheet. It is applicable to spatial data representing line objects managed in the rectangular lattice such as road networks and gas pipelines. The procedure was empirically applied to the map sheets containing the road network data of Tokyo 23-ku area, Japan, and successfully combined the map sheets into seamless spatial data.

Advantages of our approach over existing methods are summarized as follows.

1) Our method uses both global and local transformations to merge map sheets. Consequently, it works for maps distorted in either scales; unlike global methods, it produces exactly seamless maps; unlike local methods, it keeps unnatural deformation of maps to a minimum.
2) The rectangular boundaries of map sheets effectively work as a key information to extract end nodes. They greatly contribute toward extracting correct nodes, and consequently, improve the efficiency of the whole process.

3) Our method does not use attribute information of line objects including identification numbers, which provides its wide applicability. This property makes it applicable to polygon data whose boundaries often do not have attribute information; we can merge map sheets containing polygon objects by treating their boundaries as line objects.
4) Unlike existing local methods, the transformation function works not only near the map boundaries but also over the whole map sheet. This provides smooth transformation of spatial objects.
5) The map transformation keeps the topology of original lines. This property is indispensable especially to line objects such as road networks where the topology is of great importance.

One of the characteristics of our method is, as mentioned above, that it uses the rectangular boundaries of map sheets as a key information to extract end nodes. This is because spatial data are usually managed in rectangular lattice. However, the method can be applied to other systems of data management such as the longitude-latitude system; we can replace the linear function used in estimation of map boundaries by curvilinear functions to treat various systems. In addition, our method works even if the map sheet system is not known. Though the accuracy of end node extraction decreases, we can still merge map sheets by interactively checking end nodes extracted.

Our method can be extended further, because it does not make full use of information provided by spatial objects. For example, if lines have attribute information, it will drastically improve the accuracy of the procedure. In road network data, names of major streets are often recorded as an attribute, which are quite useful for matching node extraction. Even the data of road width would be helpful in finding node pairs. The shape of line objects also gives a clue to find matching nodes, because lines have to be combined smoothly at the map boundary. Though it was sufficient in the empirical study to use only the locational information of end nodes, taking additional information into account is an important subject for future research.

References

Beard, M. K., Chrisman, N. R., 1988: Zipper: a localized approach to edgematching. *American Cartographer* **15**: 163-172.

Nakaya, T., 1997: Statistical inferences in bidimensional regression models. *Geographical Analysis* **29**: 169-186.

Petersohn, C., Venderohe, A., 1982: Site-specific accuracy of digitized property maps. *Auto-Carto* **5**: 607-619.

Sato, T., Sadahiro, Y., and Okabe, A., 2000: An interactive method of detecting and visualizing the property of spatial data for joining spatial data units. *Papers and Proceedings of Geographic Information Systems Association* **9**: 399-402.

Sato, T., 2001: An interactive method for joining spatial data units. *Master's Thesis*, Department of Urban Engineering, University of Tokyo.

Tobler, W. R., 1994: Bidimensional regression. *Geographical Analysis* **26**: 187-212.

White, M., Griffin, P., 1985: Piecewise linear rubber sheet map transformation. *American Cartographer* **12**: 123-131.

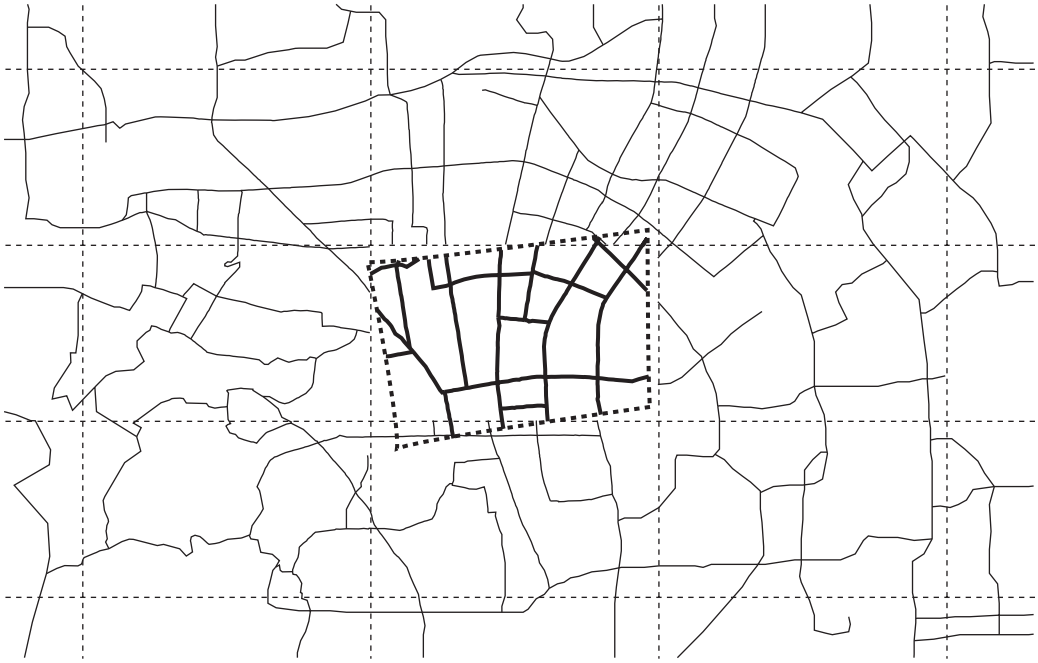


Figure 1

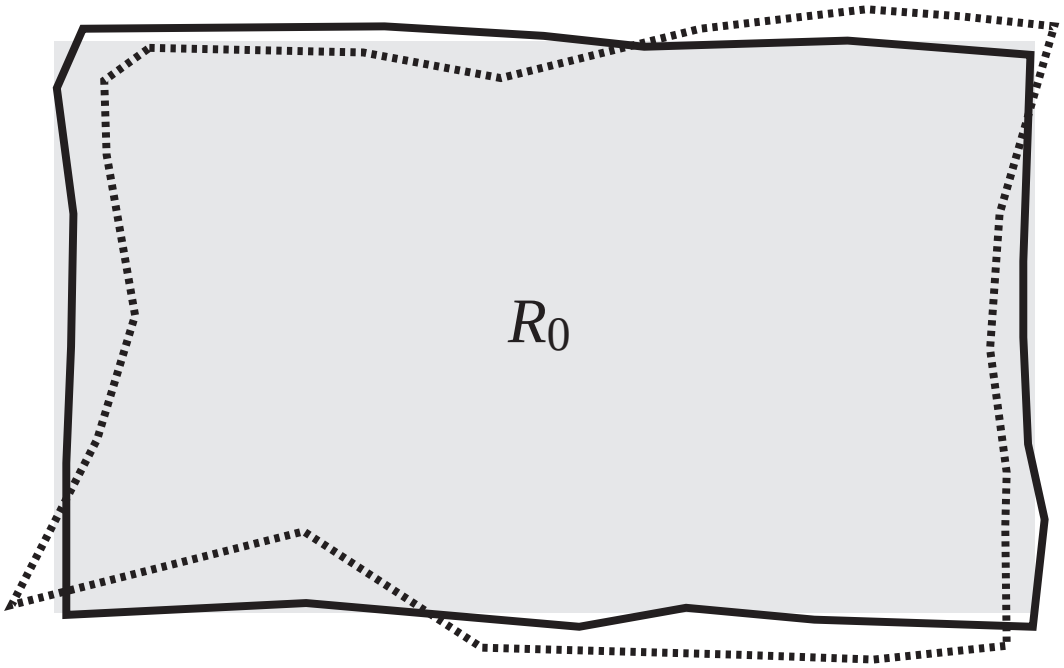


Figure 2

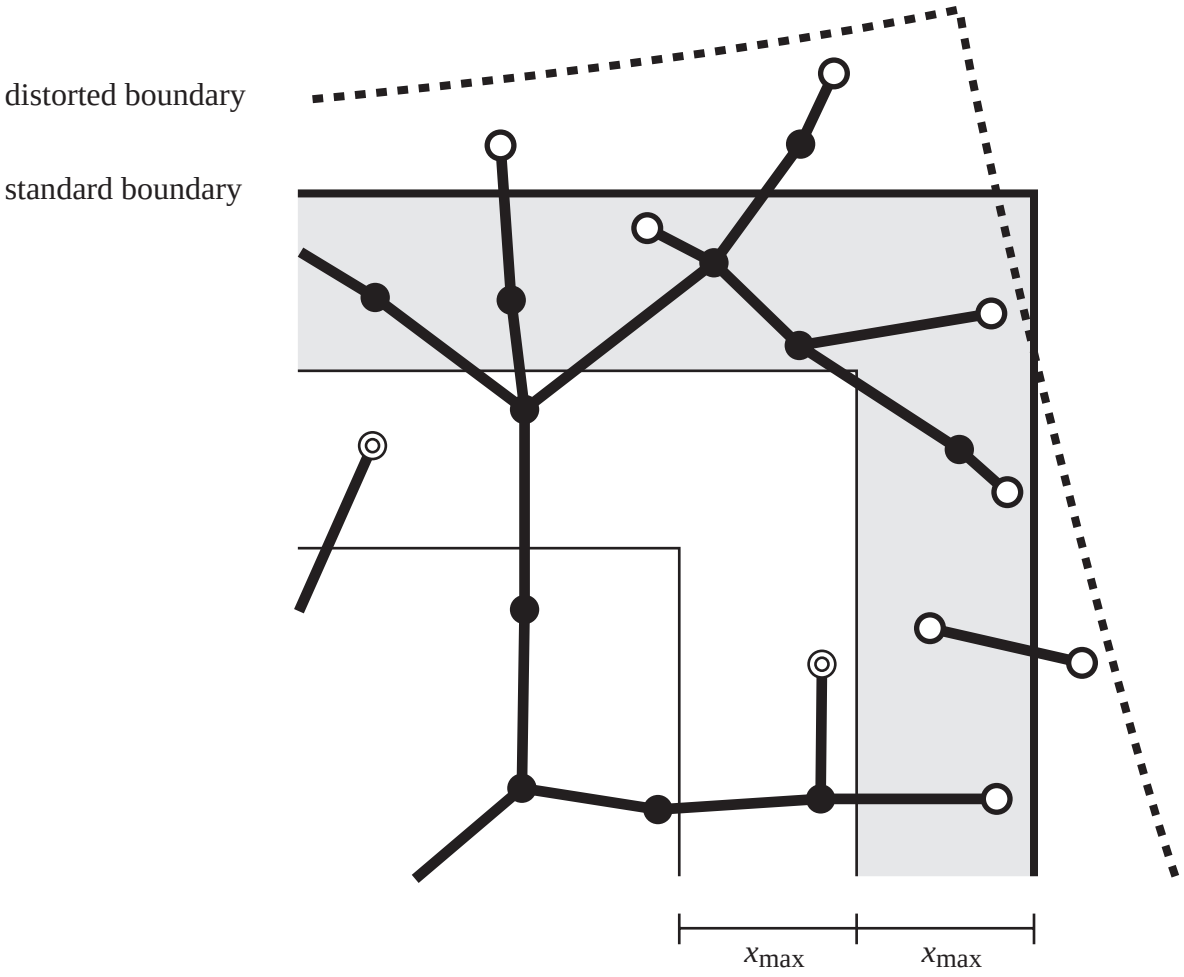
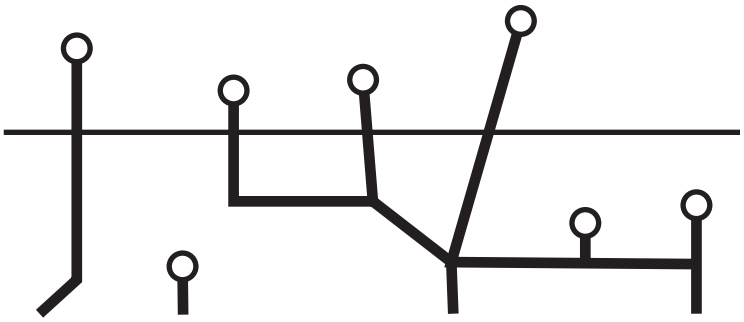
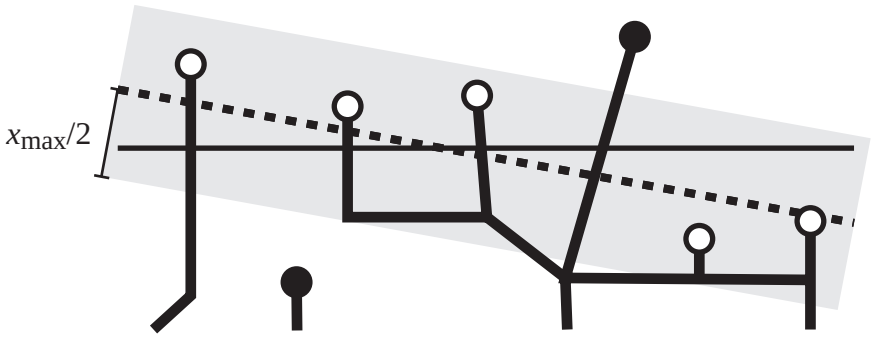


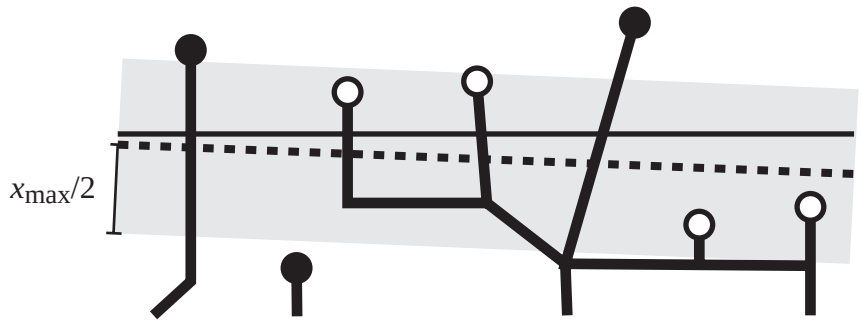
Figure 3



(a)

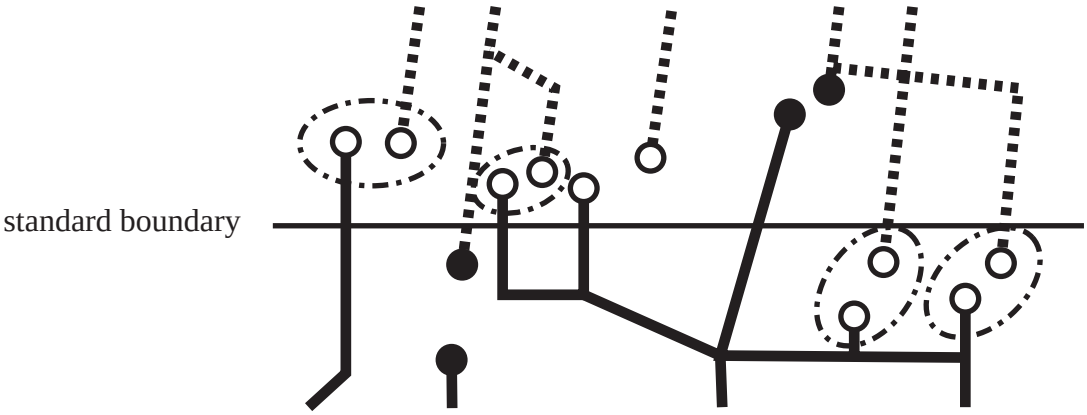


(b)

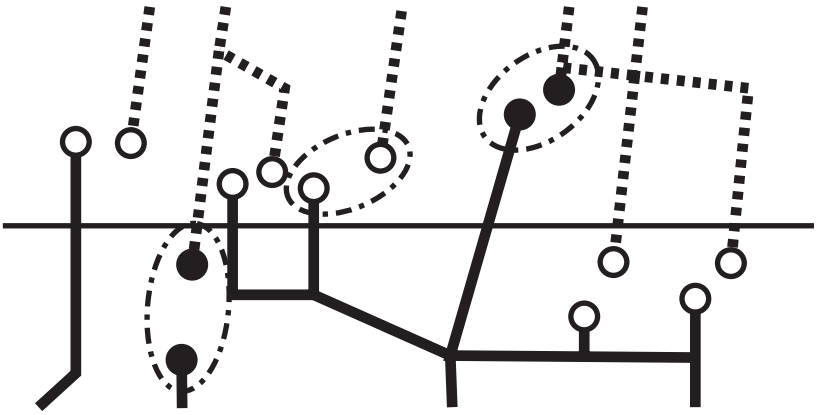


(c)

Figure 4



(a)



(b)

Figure 5

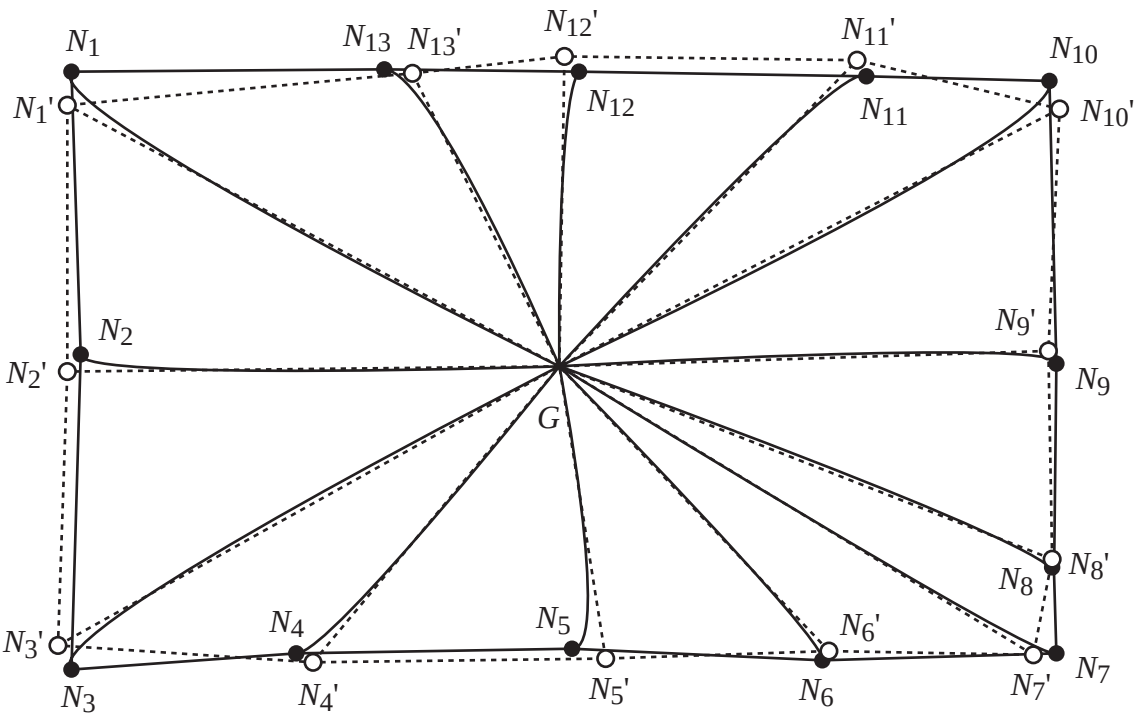

N_1
N_{13}
N_{13}'
N_{12}'
N_{11}'
N_{10}
N_1'
N_{12}
N_{11}
N_{10}'
N_2
N_9'
N_2'
N_9
G
N_8'
N_8
N_4
N_5
N_6'
N_3'
N_7
N_3
N_4'
N_5'
N_6
N_7'

Figure 6

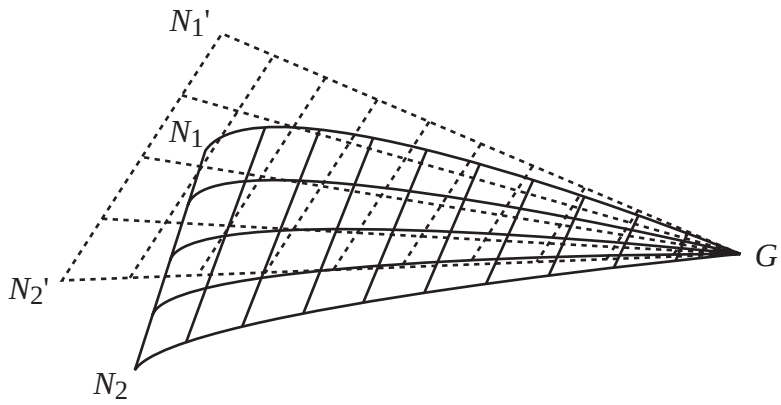
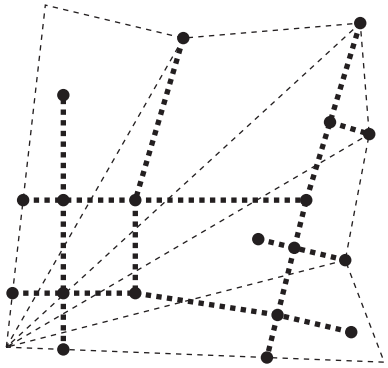
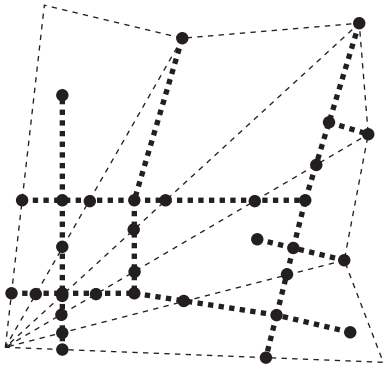


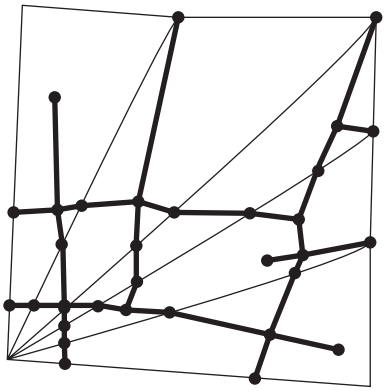
Figure 7



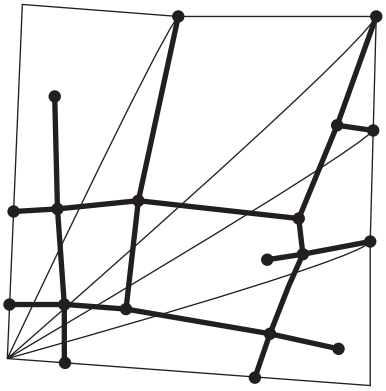
(a)



(b)



(c)



(d)

Figure 8

Figure 9

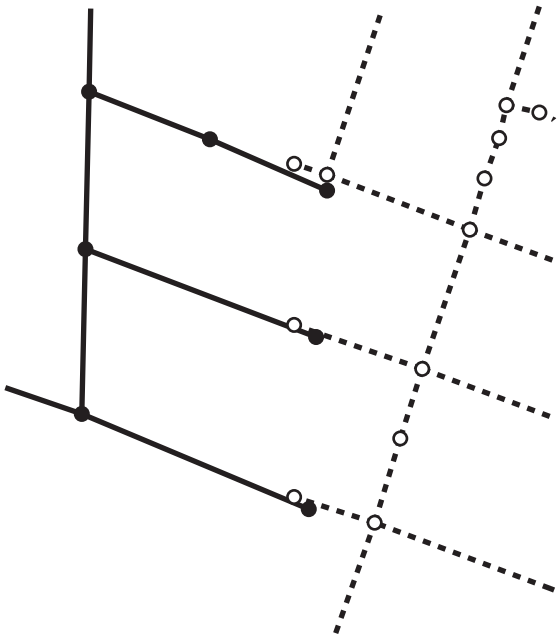


Figure 10

Figure 11

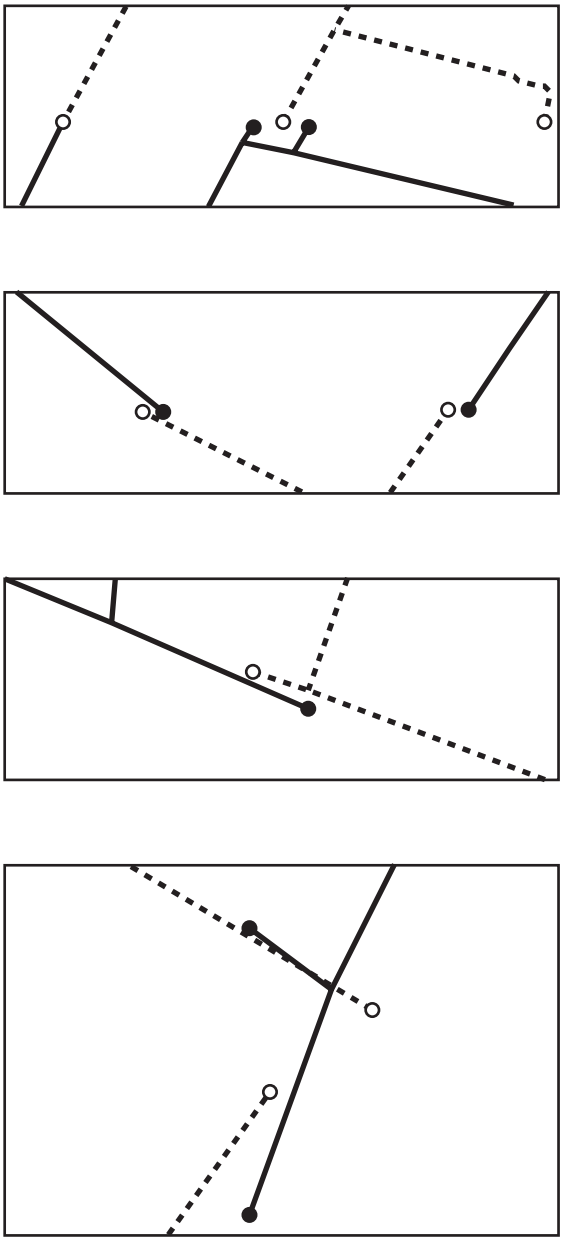


Figure 12

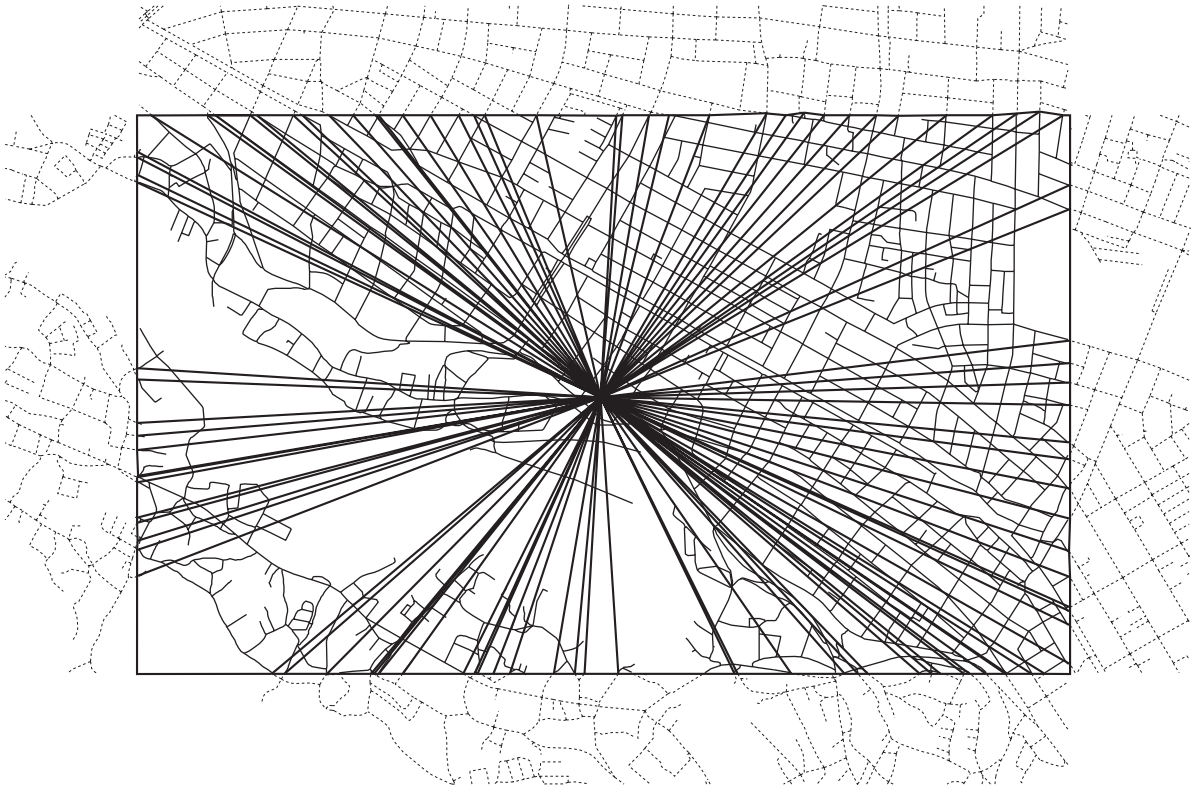


Figure 13

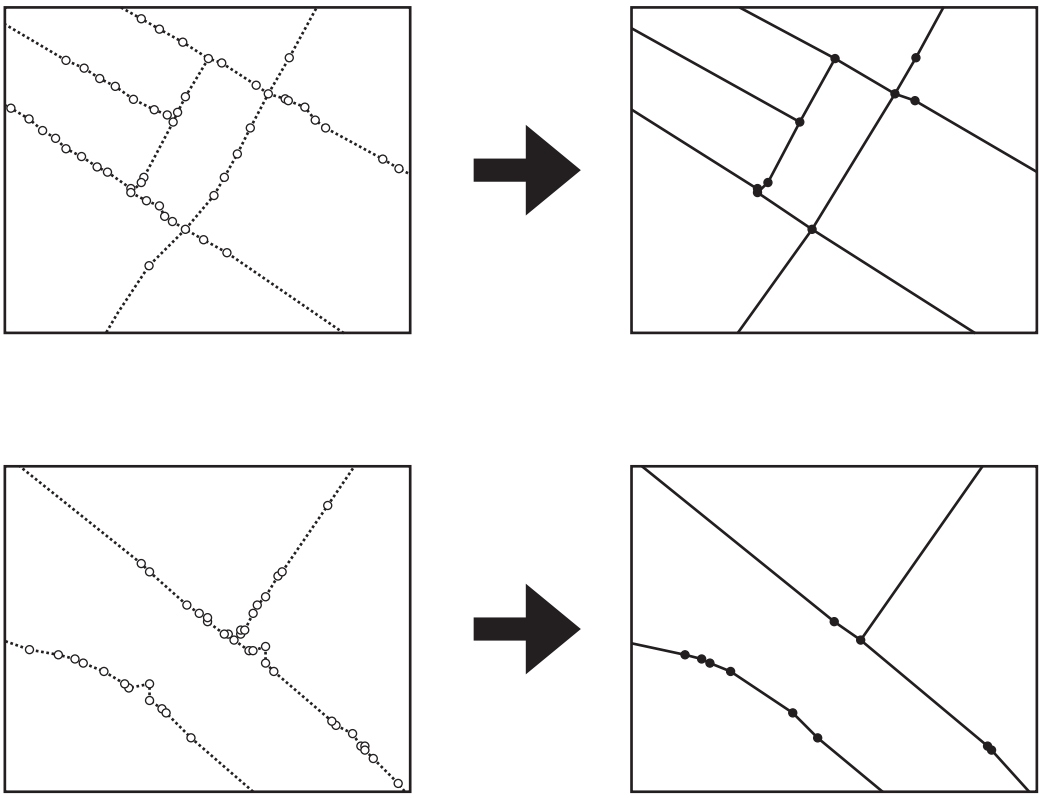


Figure 14

Figure 15