Fisheye Map Using Stroke-Based Generalization for Web Map Services

Daisuke YAMAMOTO†, Member, Masaki MURASE†, Nonmember, and Naohisa TAKAHASHI†, Member

SUMMARY A fisheye map lets users view both detailed and wide areas. The Focus+Glue+Context map is a fisheye map suited for Web map systems; it consists of a detailed map (i.e., Focus), wide-area map (i.e., Context), and an area to absorb the difference in scales between Focus and Context (i.e., Glue). Because Glue is compressed, the road density is too high to draw all of the roads in this area. Although existing methods can filter roads to draw, they have problems with rendering the road density and connectivity in Glue. This paper proposes an improved method to filter roads in Glue by applying a generalization method based on weighted strokes. In addition, a technique to speed up the proposed method by using a weighted stroke database is described. A prototype Web map system with a high level of response was developed and evaluated in terms of its connectivity, road density, and response.

key words: fisheye view, web map service, stroke

1. Introduction

Recent years have seen the wide use of Web map services such as Google Maps and OpenStreetMap. With these general Web map services, users can view the area of interest by expanding, shrinking, and scrolling the map. However, these Web map services have the drawback of a high cognitive cost if users want to see two or more distant points and a wide-area map to show the relationship between these points at the same time.

In order to solve this problem, Focus+Context-type fisheye-view maps [1]–[3] have been proposed that are based on cognitive mapping [4]–[6]. Because these fisheye-view maps need to deform the entire screen, they cannot apply the “tiling method,” which is employed in common Web map services, and thus are difficult to implement. Therefore, we previously proposed the development of Focus+Glue+Context-type fisheye-view maps [7], [8] (i.e., Focus+Glue+Context map), as shown in Fig. 1. This map consists of three areas: an enlarged map of the region of interest (i.e., Focus), a wide view of the surrounding area (i.e., Context), and a boundary area to absorb the distortion due to the scale difference between Focus and Context (i.e., Glue).

Although Glue has to be generated dynamically, the Focus+Glue+Context map has the advantage that the existing tiling method can be applied to Focus and Context because these areas have no distortion. Therefore, Focus+Glue+Context maps can be developed as a Web map service with high response.

However, because Glue is compressed, not many roads can be drawn in Glue. Although we need a way to efficiently filter roads in Glue, existing methods have problems in terms of bias of the road density and connectivity.

The purpose of this study was to develop an improved filtering method of roads in Glue while maintaining a high response speed to enhance the visibility and connectivity in this area. Specifically, we introduce the concept of strokes to select roads in Glue. The proposed method can reduce the number of roads along both the angular and radial directions by considering the connectivity and weight of strokes, and controlling the number of roads at both sides of Glue. Moreover, we developed the prototype system that realizes a high response by constructing a weighted stroke database, and evaluated in terms of its connectivity, road density, and response.

2. Related Work

Thomson and Richardson introduced the concept of a stroke into road generalization using road geometry [10]. Strokes are network elements that combine both functional importance and perceptual significance in map generalization and are derived by introducing the “good continuation” principle of perceptual grouping into networks. Strokes are constructed and ordered according to predefined rules, and the selection of roads is then simplified as the selection of strokes with a higher order. Thomson and Richardson’s stroke-based generalization orders strokes based on the stroke length and road class. Zhang presented a method
for selecting salient roads based on connection analysis [11]. This method counts the number of connections at each junction to serve as a parameter indicating the association between salient roads. Several strategies have been proposed to combine connection criteria with the road length and road attributes as a way to order roads corresponding to their relative importance to the road network generalization. An algorithm for the elimination of arcs in road maps was presented based on information theory [10]. This function is computed based on the length between any two information points. Based on the information points and similarity function, the entropy and equivocation of the road map can be computed. The conditional probabilities are derived from a similarity function, and mutual information is derived from the relation between the similarity and conditional probability. By using information theory, the elimination algorithm computes how many roads can be presented on a map by maximizing the amount of useful information. Yungang et al. proposed the selective omission of streets based on the ratio of area to length of roads that surrounds the area, which they called the mesh density [12]. A narrow road interval is a location where the mesh density is higher. Omitting such roads enables a small-scale road network to be created. Jinang and Claramunt proposed a generalization model to select characteristic streets of an urban street network [13]. Their model retains the central structure of a street network and relies on a structural representation of a street network called a connectivity graph. They also introduced local and global measurements to qualify the status of each vertex within the graph. Murase [14] proposed an online generalization method that uses a stroke database for Web searches. This method calculates the importance of strokes based on the number of facilities at the wayside of the stroke and draws the important strokes. Haunert [15] proposed a map transformation method based on the convex quadratic program in order to draw Focus. Although this system has to update the entire map dynamically, it achieves a more distortion-free map than the proposed method. Dijk [16] proposed a schematized focus maps. This method improves the visibility of Context area by connecting a Focus with schematized maps.

3. Focus+Glue+Context Map and Its Problems

A Focus+Glue+Context map consists of Focus (large-scale area), Context (small-scale area), and Glue (area that connects Focus and Context), as shown in Fig. 2. The Focus–Glue boundary refers to the border separating Glue and Focus, while the Glue–Context boundary refers to the border separating Context and Glue.

Figure 3 shows the layer structure of a Focus+Glue+Context map: Glue, Focus, and Context in that order. The Glue layer has to draw a distorted map by using the relocation function to absorb the scale difference between Focus and Context. Therefore, map images of Glue have to be generated in the server and sent to the client every time a user controls the map. On the other hand, because map images of the Focus and Context layers have no distortion, they can be displayed by using the same tiling method employed by existing Web map services. Because existing Focus+Context-type fisheye-view maps have to update all layers, the Focus+Glue+Context map has advantages in terms of the response time and network cost.

Roads in Glue are transformed by the relocation function, as shown in Fig. 4. The horizontal axis is for the distance from the center of Focus before transformation. The vertical axis is for the distance from the center of Focus after transformation. The vertical axis is for the distance from the center of Focus after transformation. The relocation function uses a Bezier curve in order to improve visibility by smoothly connecting curves.

However, Glue has the problem that the scale along the
radial direction is too small. Figure 5 shows the scales along the radial and angular directions. In particular, because the scale along the radial direction of Glue is smaller than that of Context and Focus, the road density is too high to draw all roads. On the other hand, the scale along the angular direction is larger than the scale along the radial direction and changes linearly from the scale of Focus to the scale of Context.

The scales along the radial and angular directions of $S_{GR}$ and $S_{GA}$, respectively, are determined by the following formulas:

$$S_{GR} = \frac{kR_C - R_F}{kR_C - R_F}S_C$$  \hfill (1)$$

$$S_{GA} = S_C \sim S_F$$  \hfill (2)$$

where $S_F$ is the scale of Focus, $S_C$ is the scale of Context, $R_F$ is the radius of the Focus–Glue boundary, and $R_C$ is the radius of the Glue–Context boundary. $a \sim b$ means that the value changes linearly from $a$ to $b$. The scale ratio, $k$, is defined using the scale, $S_F$, of Focus and the scale, $S_C$, of Context, as follows:

$$k = \frac{S_F}{S_C}$$  \hfill (3)$$

For example, for the map in Fig. 2, $S_{GR}$ is 1/90000 when $R_F = 200$, $R_C = 250$, $S_F = 1/15000$, and $S_C = 1/30000$.

In order to solve the problem of the radial direction, the existing method [8] only draws the roads along the radial direction based on the concept that roads in Glue only have to connect those in Focus and Context. This method is effective at reducing the number of roads along the angular direction and improving visibility. However, it has the following problems.

Problem 1 (Bias of road density) As shown in Fig. 6, the existing method has a bias in road density: the road density is higher near Focus than near Context. Because the existing method reduces the number of roads based on the path being followed, some roads in Focus cannot reach Context or are fused into one road when the scale difference between Focus and Context is large.

Problem 2 (Connectivity) The existing method has a connectivity problem: because the number of roads near Context is low because of Problem 1, some paths from Focus to Context need to detour. In other words, the length of a path from Focus to Context may be much longer than the length of the shortest path in the real world.

Problem 3 (Response time) Because the proposed system has to work in real time, applying complex algorithms is difficult.

In order to solve these problems, we propose the following approach. In order to solve Problem 1 and Problem 2, we propose the Glue generalization method based on weighted strokes. In order to solve Problem 3, we propose speeding up the method by using a weighted stroke database.

4. Proposed Method

In this section, we describe the proposed method.

4.1 Stroke Generation

In this paper, a “stroke” refers to a list of roads based on the concept of “good continuation” from the principle of perceptual grouping [9], [10]. We store strokes in a weighted stroke table, as given in Table 1. We generate strokes from the road network of OpenStreetMap.

We describe the algorithm to generate strokes.

The following is defined for each link $L$ in the road network shown in Fig. 7.

A road network is represented by an intersection (node) and the road (link) that connects the two intersections. The length of a link is along the path of the road between the intersections. A stroke is a sequence of links that satisfies conditions (a) to (c) described below. Intuitively, it is a road that follows the road and does not necessarily coincide with

![Fig. 5](image-url) Scale problems in Glue: (a) schematic, (b) scale along the radial direction, and (c) scale along the angular direction.

![Fig. 6](image-url) Bias in Glue. The road density is higher near Focus than near Context.

<table>
<thead>
<tr>
<th>Table 1</th>
<th>Weighted stroke table.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Column</td>
<td>Type</td>
</tr>
<tr>
<td>Stroke_id</td>
<td>Int</td>
</tr>
<tr>
<td>Stroke_line</td>
<td>MultilineString</td>
</tr>
<tr>
<td>Stroke_length</td>
<td>Double</td>
</tr>
<tr>
<td>Stroke_weight</td>
<td>Double</td>
</tr>
</tbody>
</table>
an administrative road (for example, National Highway 1). For example, stroke S3 in Fig. 8 (b) is a sequence of links L5, L10, and L17, which are shown in Fig. 8 (a). This stroke is represented as S3 = (L5, L10, L17).

The link set LS follows the links of the link L, the link set LSi follows the links of a link Li that is included in LS, and LS and LSi are in the same road class as L. Table 2 presents the road class, which was classified by referring to OpenStreetMap.

- LS = \{L1, L2, ..., Lk\}
- LS = \{L1, L2, ..., Li, ..., Lk\}
- len(Li): length of link Li
- angle(LA, LB): angle between links LA and LB
- minAngle(LA, S): \{angle(LA, LB), LB \subset S\}

Here, we define the following rule to detect a continuous link as the stroke S:

(a) IF LS = \{L1\} THEN L, L1 belong to the same stroke S.
(b) IF LS = \{L1, ..., Lk\}(k ≥ 2) and angle(L, Li) < δ and minAngle(L, LS) = Li THEN L, Li belong to same stroke S.
(c) IF LS = \{L1, ..., Li\}(k ≥ 2) and len(Li) < Δ and angle(L, Li) < δ and minAngle(L, LS) = Li THEN L, Li belong to same stroke S.

where Δ and δ are heuristic values. In this paper, we define δ = 45°. The value of Δ was set to 10 meters. Figure 7 and Fig. 9 shows a sample stroke detected by the above rules. Links L, Li, and Lij belong to the same stroke.

4.2 Weighted Stroke

In general, the weight of a stroke should be defined by considering various aspects such as the road class, stroke length, and purpose of use. However, in order to simplify the problem, we use only the stroke length to define the weight. This is because we have heuristics that show that many long roads may be important, such as national roads.

Here, let the weight WS of the stroke S be as follows:

\[ W_S = \text{len}(S) \]  

where \text{len}(S) is the length of the stroke S. The length of the stroke is the sum of the lengths of all links comprising the stroke. The stroke S and its weight W_S are stored in a weighted stroke table, such as Table 1.

4.3 Glue Generalization

Our proposed Glue generalization method is based on weighted strokes. First, the number of roads to draw is determined based on the scale along the angular direction in Glue. Next, roads are filtered based on the weighted strokes intersecting each boundary line in order to be less than the number of roads determined above. In addition, the connectivity between Focus and Context needs to be considered.
4.3.1 Number of Strokes

Our proposed method determines the number of roads in Glue based on the road density.

First, we define Glue’s road density $D_G$ as follows:

$$D_G = \frac{A_S}{A_G}$$ (5)

where $A_G$ is the onscreen area (square measure) of Glue and $A_S$ is the onscreen area of the stroke lines to draw. The area of the stroke $A_S$ is not the area of the road in the real world; it is the area occupied by the lines drawn (having a width). $A_S(k)$ is the area of top-k strokes drawn in Glue based on weight of stroke $W_S$. Therefore, we calculate the value by using the following function:

$$A_S(k) = \sum_{i=0}^{k} \text{len}(S_i)width(S_i)$$ (6)

where $width(S_i)$ is the width of the line for the stroke in Glue and $k$ is the number of strokes to draw. $k$ is defined in the following function by using the threshold $\beta$:

$$n = \arg \max_k \left( \frac{A_S(k)}{A_G} < \beta \right)$$ (7)

Figure 10 shows how the number of selected roads changes depending on the size of Glue. Thus, the density of roads in Glue is set to be constant.

4.3.2 Stroke Selection in Glue

The stroke set $S_{\text{draw}}$ is calculated with the following algorithm by using the number of strokes $n$ described above:

Step 1 $S_{fG}$ is a stroke set intersecting with the Focus–Glue boundary.

Step 2 $S_{Gc}$ is a stroke set intersecting with the Glue–Context boundary.

Step 3 $S_{\text{candidate}}$ is the top $N$ strokes from $S_{fG}$ and $S_{Gc}$ based on the weight of each stroke.

Step 4 For each stroke $S'$ included in $S_{\text{candidate}}$, if the end of $S'$ is within Glue, the strokes to touch the end of $S'$ are added to $S_{\text{candidate}}$.

Step 5 $S_{\text{draw}}$ is a set of deduplication strokes contained in $S_{\text{candidate}}$.

Step 4 is effective at improving connectivity because some strokes are interrupted when only Steps 1–3 are used.

4.3.3 Relocation and Drawing

Finally, we can draw the selected strokes by using the relocation function and existing method [8].

5. Prototype System

We developed a prototype system as a Web map service. We adopted the Java language for the server, and JavaScript for the client (Web browser). We used Leaflet\(^1\), which is a map library for JavaScript. For the server, we used the map data of OpenStreetMap\(^2\) and developed a database by using PostgreSQL and PostGIS\(^3\).

5.1 System Structure

The proposed system consists of a stroke generator and Glue server, as shown in Fig. 11.

The stroke generator generates strokes, calculates these weights, and stores them to the stroke database. The scheme of a stroke table is as shown in Table 1. The glue server determines the number of strokes, filters strokes, and generates the Glue image by using the relocation function.

The reason for performing preprocessing to generate a stroke is to improve response time. To generate a stroke, it is necessary to load roads within a sufficiently wide area from the road database and expand to the memory and search these roads. This leads to a high computational cost. If the stroke database is not used, the calculation process is

\(^1\)http://leafletjs.com/
\(^2\)http://www.openstreetmap.org/
\(^3\)http://postgis.refractions.net/
performed for each request and the response time deteriorates remarkably. In the proposed method, it is not necessary to calculate each time by preprocessing and store the result into the database, which makes it possible to improve the response time.

Figure 12 shows the interface of the proposed system. Because this system was developed as a Web service, users can use the system via a Web browser. The user interface of the prototype system is the same as that for the existing method [8]. This interface enables users to control the positions of Focus and Context, the radii of Focus and Glue, the scales of Focus and Context, and the generation method of Glue. Generation methods of Glue include the existing method (following road) and proposed method.

5.2 Implementation of Glue Server

We developed a Glue server by using the proposed method. This system can receive HTTP requests with parameters from Web clients and return PNG image files of Glue, as well as the results of the existing method [8]. The parameters of HTTP requests are as follows:

- centerLgLat: latitude and longitude of the center of Focus
- focus_zoom_level: scale level of Focus
- context_zoom_level: scale level of Context
- glue_inner_radius: radius of the Focus–Glue boundary
- glue_outer_radius: radius of the Glue–Context boundary

5.3 Implementation of Focus+Glue+Context Map

The structure of the Focus+Glue+Context map was developed and compared with the results of the existing method [8]. Although we used static map images for Focus and Context in this study, the proposed system supports dynamic images drawn by stroke-based generalization for Focus and Context maps.

6. Experimental Results

This paper mainly describes the Glue generation method for Focus+Glue+Context maps. We evaluated Glue by comparing our results with those of the existing method (following path) in terms of the road density, connectivity, and response time. We used the road network of OpenStreetMap and set the central part of Nagoya City, Japan, as the target area.

6.1 Evaluation of Connectivity

In general, because the Glue area is compressed, we have to reduce the number of roads in order to improve visibility. Moreover, appropriate routes from Focus to Context should be displayed by priority in Glue. Although it is difficult to define the appropriateness of routes in Glue, we adopted the path length from the center of Focus to any point in Context as a metric.

We define the metrics of connectivity of Glue as follows: For any Focus $f$ and any point in Context $p$, $Len(f, p)$ is the length of the shortest path from $f$ to $p$ by using all roads (including non-displayed roads). $Len(f, p)$ calculates all links on the road network, including roads with unscreened scales. $GLen(f, p)$ is the length of the shortest path from $f$ to $p$ by using only displayed roads in the Focus+Glue+Context map. The connectivity metric $Con(f, p)$ is defined as follows:

$$Con(f, p) = \frac{GLen(f, p)}{Len(f, p)} \quad (8)$$

Similarly, the connectivity metric $Con'(f, p)$ is defined as follows. $Con'(f, p)$ is the speed limit for each road. $GLen'(f, p)$ and $Len'(f, p)$ define the travel time from dividing the path length by the speed limit.

$$Con'(f, p) = \frac{GLen'(f, p)}{Len'(f, p)} \quad (9)$$

We further discuss the significance of the above formula below. Because the generalized road network is less detailed than the original road network, the lengths of some paths from Focus to Context may be much longer because of detours. Therefore, we used the ratio of path lengths as a connectivity metric.

We evaluated the connectivity by comparing the results of the existing and proposed methods according to the connectivity metric $Con(f, p)$. The experimental conditions are as follows: the radius of the Focus–Glue boundary is 200 pixels, the radius of the Glue–Context boundary is 400 pixels, and the target points $p$ intersect with a circle having a radius of 500 pixels. These experimental conditions set Glue’s width to as large as 200 pixels so that the evaluation of Glue is easy; however, it is within the range of the general settings considered in our system. Table 3 presents two cases. Case 1 is for pedestrian navigation. In this case, we use a large-scale map and the connectivity metric $Con(f, p)$, which only
Table 3  Experimental conditions for connectivity evaluations. The scale level is the zoom level defined in Open Street Map. The scale differs depending on the latitude. Scale 16 shows a map of approximately 1:8000 and Scale 14 shows a map of approximately 1:35000.

<table>
<thead>
<tr>
<th>Case</th>
<th>Scale level in Focus</th>
<th>Scale level in Context</th>
<th>Cost function</th>
<th>Number of points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case 1</td>
<td>16</td>
<td>15</td>
<td>Distance only</td>
<td>200</td>
</tr>
<tr>
<td>Case 2</td>
<td>14</td>
<td>14</td>
<td>Distance and speed limit</td>
<td>307</td>
</tr>
</tbody>
</table>

Fig. 13  Connectivity evaluation results for Case 1. The vertical axis is for the connectivity metric $\text{Con}(f, p)$. The horizontal axis is for point IDs in order of $\text{Con}(f, p)$ with the existing method.

Fig. 14  Connectivity evaluation results for Case 2.

The vertical axis is for the connectivity metric $\text{Con}(f, p)$; we plotted all of the results for each point $p$ in order. In order to improve the visibility of the graph, points are displayed in ascending order of $\text{Con}(f, p)$ with the existing method. A smaller value for $\text{Con}(f, p)$ is better. For most of the points, the proposed method showed equal or better connectivity than the conventional method. These results suggest that the proposed method selected the appropriate route. In case 1, 60% of points $p$ improved, and the path length was reduced by up to 23% (4.9% on average). In case 2, 24% of points $p$ improved, and the path length was reduced by up to 18% (3.7% on average).

6.2 Evaluation of Road Density

The existing method has a problem with road density, as described in Problem 1. Specifically, the road density near the Focus–Glue boundary is much greater than that near the Glue–Context boundary when the scale of Focus is much large than that of Context. Therefore, we compared the proposed and existing methods in terms of the road density along both the radial and angular directions, as shown in Fig. 5 (a).

We counted the strokes intersecting circles and radial lines from the center of Focus in order to determine the road density. For the angular direction, we counted strokes intersecting 10 equally divided circles between the Focus–Glue and Glue–Context boundaries, as shown in Fig. 16 (a). We calculated the scales along the angular direction for each circle as follows:

$$\frac{\text{number of intersected strokes}}{\text{length of circle}}$$  \hspace{1cm} (10)

For the radial direction, we counted strokes intersecting radial lines drawn from the center of Focus at 30° intervals, as shown in Fig. 16 (b). We calculated the scales along the radial direction for each degree interval as follows:

$$\frac{\text{number of intersected strokes}}{\text{width of Glue}}$$  \hspace{1cm} (11)

The experimental conditions were as follows: the central area of Nagoya City, Japan, was set as the target area; the map size was 1024 pixels × 1024 pixels; the radius of the Focus–Glue boundary was 200 pixels; the radius of the Glue–Context boundary was 400 pixels; the scale level of Focus was 14; the scale level of Context was 13; and $\alpha = 0.3$.

Figure 17 shows the results when the road density was measured along the angular direction. Although the road densities of the proposed and existing methods were similar at the Glue–Context boundary (400 pixels), the existing method had a density that was three times that of the proposed method at the Focus–Glue boundary (200 pixels). The Focus–Glue boundary was 8.3 times denser than the Glue–Context boundary with the existing method. Figure 18 shows the results when the road density was measured along the radial direction. The densities were almost the same with both methods regardless of the angle.
Fig. 15 Example connectivity result for case 1. The green line is the shortest path from the center of Focus to point \( p \) in Context. (a) Shortest paths using all roads. (b) Shortest paths using roads filtered by the existing method (following path). (c) Shortest paths using roads filtered by the proposed method.

Fig. 16 How to measure road density. a) We counted strokes intersecting red circles for the angular direction. b) We counted strokes intersecting red lines for the radial direction.

Fig. 17 Road density along the angular direction.

Fig. 18 Road density along the radial direction.

These results suggest that the proposed method can control roads effectively near the Focus–Glue boundary. Although the existing method only controls the number of roads at the Focus–Glue boundary, the proposed method can control the number of roads at both the Focus–Glue and Glue–Context boundaries.

Figure 19 shows the comparison of the proposed and existing methods. Even though the existing method has several roads close to the Focus-Glue boundary, the proposed method controls the number of roads close to the Focus-Glue boundary.

6.3 Evaluation of Response Time

Finally, we evaluated the response time of the proposed system.

Because the proposed system works with Web map services, the response time is a critical issue. Because the Glue server is the bottleneck of the proposed system, we measured its response time.

The experimental conditions were as follows. The central area of Nagoya, Japan, was set as the target area. The map size was 1024 pixels × 1024 pixels. The radius of the Focus–Glue boundary was 200 pixels. The radius of the Glue–Context boundary was 400 pixels. We measured response time for each scale of Focus and Context.

Figure 20 shows the execution time of the Glue server for each scale. The proposed system was slower than the existing system by 0.2–0.4 s. The bottleneck of our system was the query time of the stroke database. We expect that
speed-up techniques for the database, including replication and a NoSQL database such as BigTable†, will be effective with our system because this is not a very complex query. This system does not use these speed-up technologies.

6.4 Evaluation of User Factors

We conducted an evaluation experiment on the visibility of the Focus+Glue+Context map. We showed this map to the subjects and evaluated their visibility using questionnaires.

The target areas were 5 arbitrary areas of Nagoya city in Japan. The zoom level of Focus was 16, the zoom level of Context was 14, the diameter of Focus was 300 pixels, and the width of Glue was 100 pixels. The subjects were 10 college students. The comparison method was a stroke-based method, a conventional method (following path), and drawing all roads as a baseline. The questionnaire was comprised of the following seven items:

1. Q1. Is the visibility of Glue good?
2. Q2. Is the visibility of Focus good?
3. Q3. Is the visibility of Context good?
4. Q4. Is Glue’s road connectivity good?
5. Q5. Is the connection between Focus and Glue boundaries good?
6. Q6. Is the connection between Glue and Context boundaries good?

Experimental results are shown in Fig. 21. Regarding the visibility of Focus and Context (items Q2 and Q3), there is no significant difference between the proposed method and the conventional method at a significance level of 5%. As the drawing method of Focus and Context has not been changed, it is natural that there is no significant difference. On the contrary, with regard to Q1 (visibility of Glue), the proposed method (4.36) is superior to the conventional method (4.02) at a significance level of 5%. Regarding Q4 (connectivity of Glue), the proposed method (3.5) is superior to the conventional method (2.96) at a significance level of 5%. Similarly, for items Q5 and Q6, the proposed method is superior at a significance level of 5%. These results are consistent with the experimental results of road density and connectivity described in the previous section. Therefore, the visibility and connectivity of proposed method are improved compared to existing methods.

7. Conclusion

In this paper, we propose an improved generalization method to filter roads in Glue based on weighted strokes. Moreover, we describe a technique to speed up the proposed method by using a weighted stroke database. Although the existing method only reduces the number of roads along the angular direction, the proposed method can reduce the number of roads along both the angular and radial directions by considering the connectivity and weight of strokes. More-
over, although the existing method only controls the number of roads at the Focus–Glue boundary, the proposed method can control the number of roads at both the Focus–Glue and Glue–Context boundaries.

We developed a prototype Web map system with a high level of response and evaluated the proposed method in terms of the connectivity, road density, and response. We confirmed that the proposed method improves the connectivity and road density of Glue.

Although the proposed method adopts a simple stroke-based generalization, in the future we will consider other more advanced generalization methods [17]. We will also study speed-up techniques for databases such as BigTable in order to improve the response time.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Numbers JP26330136, JP25700009 and SCOPE.

References