A Method to Visualize Numerical Data with Geographical Information Using Feathered Circles Painted by Color Gradation

Kenichi Iwata   Mariko Sasakua
Department of Computer Science, Okayama University
Tsushima-Naka 3–1–1, Okayama, 700-8530, Japan
{iwa, sasakura}@momo.cs.okayama-u.ac.jp

Abstract

Some kind of data has its value and subsidiary data of geographical information. This paper presents a method to visualize this kind of data. A typical example of the data is a value with location information that the value is measured. It helps viewers understanding them to show both of the values and geographical information at the same time.

The method we propose plots data on a map using circles painted by color gradation which edges are feathered. We also show algorithm and implementation of the method. The result of plotting data on a map shown in this paper is easy to understand.

Keywords: *Numerical Data, Geovisualization*

1 Introduction

There is a kind of data that is easy to understand if its geographical information is considered with. A typical case of the geographical information that the data has is the information of the location where the data is observed. For example, amount of rainfall, land prices or amount of traffic are that kind of data.

The use of positioning system such as GPS is widely spreading lately, therefore the numbers of data with geographical information is increasing. In this paper, we propose a method to visualize the data with the location information by using circles painted by color gradation of which edges are feathered.

There are several ways to obtain data with the location information, and also several ways to visualize them (table 1). The ways to obtain data are:

Case 1 measuring data on points that is on mesh intersections, and

Case 2 measuring data on specific points.

Case 1, measuring data on mesh intersections, is good to obtain enough data to plot data on all over a map. The data covers all area of a map, so a mesh cell can be painted by a color determined by a single value represents. Rainfall data obtained by using radar reflection is this kind of data [1, 6].

Case 2, measuring data on specific points, means we measure data only on few points that is not many enough to cover all of a map. There are some reasons we measure only on few points. A case is that the objective phenomenon is distributed spatially continuously, so we would like to measure them and plot them to cover all over a map. However because of a limitation of a way to measure data, by technological reasons or financial reasons, we cannot obtain data on mesh. In this case, we can estimate data on “empty mesh” from the limited measured data (Case 2-1).

The other case is that the objective phenomenon is not spatially continuous. It means that there are some accents of distribution of values, and other points are just empty or meaningless. For example, in a case we measure number of customers of restaurants, we choose measuring points on places where restaurants are, not at regular intervals as like we do in Case 1. In this case, we just give up to fill empty area, and plot values only on the data measuring points (Case 2-2).

When it’s possible to apply Case 1, the result is better than Case 2. However, if increasing measuring points causes increasing the cost, and it is not acceptable, we have to choose Case 2 instead of Case 1. Another case we have to choose Case 2 is that there is no way to observe data on mesh. An example of this case is counting number of tourists. A number of tourists can not be counted on 100m mesh, because tourists appear only on touristic places.

Visualization of geographical information is called Geovisualization, and it is the research area on tools and techniques for treating data with geographical information [2]. There are some existing methods to fill empty mesh on the Case 2-1. Multivate geostatics is one of the areas. It gives some researches to handle geographical data [3, 5]. Kriging is one of the famous methods to estimate spacial data, that was originally developed for estimating mineral prospect [4]. This means that you can estimate the overall distribution of

Table 1. How to obtain and plot data

	place to obtain data	how to plot data
Case 1	on every mesh intersection	plot all over a map
Case 2-1	only on specific points	plot all over a map
Case 2-2	only on specific points	plot only on measuring points

mineral resource from observation points, by using the characteristics of underground resource that is continuous spaciously.

We could apply Kriging to estimate overall spacial distribution from limited number of obtained data that is continuously distributed. However, for some data, generally, the existing Kriging method cannot be applied, since they have characteristics shown as following:

1. Some data may not have been obtained under organized situation, so it might not be enough to interpolate the overall distribution.
2. For some of data, we do not know the model for estimating missing values yet.

In Case 2-2, there is not good method to plot values on measuring points. In some cases we plot bar graph on the points.

Therefore, We propose a new method to estimate and visualize data for general use. This method works fine for Case 2. We have considered applying the method to the data obtained from Web. The proposed method is available for data which have the following characteristics.

- Obtained data can be assumed as local maximal values. Data may be represented in Web because someone thinks they are variable in some ways. We consider they may be local maximal values in mathematical terms.
- Data are continuous geographically.

To estimate and visualize such data, we propose a method to draw a colored circle that has the center on the place where the data is obtained. The circle's edge is feathered, so we call it the feathered circle. The set of feathered circles is called as the feathered graph.

A feathered graph represents data with geographical information, so it needs a map to be drawn on. We use Google Maps[1] as the map, which is a service by Google. It can be shown on normal web browsers.

2 The idea to generate feathered circles

In this section, we describe how to generate feathered circles. The main procedure to generate a feathered graph is:

1. Get data with geographical information.
2. Generate feathered circles.
3. Put feathered circles on a map.

2.1 Basic idea

As described in Section 1, the intended data are continuous geographically and the input is a set of local maximum values with coordinates in which the values are obtained. We draw one feathered circle for one value. We call the coordinates in which a value is obtained as a *source point*.

A feathered circle is drawn as the following:

- A source point becomes a center of a circle.
- The color of the center of circle is decided according to the value.
- Values around a source point are decided according to the distances from the source point. The value of the point that is nearer to the center is larger and the value of the farther point is smaller. Every point has a value lesser than the value that the center has. The color of the point is decided by only its value.
- We change the color by gradation. So, we get a feathered circle from a source point.

There are cases that a point is near more than one source point. In such case, we decide the value of the point as the maximum value calculated from each source points, because the value of each source point must be the local maximal value.

2.2 Algorithm

We generate a feathered graph as an image file by calculating a value of each dot from the data of source points. In our algorithm, a user must specify the following things:

- The function for calculating the values around a source point. We represent it as $F(point, value, d)$ where $point$ is a coordinates of a source point, $value$ is its values and d is a distance form the source point to the calculating point. If d is small, that is the source point is

[1] http://maps.google.com/

```
Get sources and sources_values.
value = 0
for i = 0 to maxw
  for j = 0 to maxh
      for k = 0 to number_of_sources
        tmpvalue = F(sources[k], sources_values[k],
        distance between sources[k] and calculating point)
        if value < tmpvalue then
          value = tmpvalue
      end
      image[i][j] = G(value)
  end
end
```

Figure 1. The algorithm to generate feathered graphs.

near, $F(point, value, d)$ must calculate bigger value than d is big, that means the source point is far from the calculating point.

- The function for deciding a color by a value. We represent it as $G(x)$ where x is a value. $G(x)$ must decide the color of the point by only the given value, x.

The input of the algorithm is a set of source points and their values. In our algorithm, we represent source points as $sources$ and their values as $sources\_values$. $maxw$ is the width of the image and $maxh$ is the height of the image. $image$ is an array of dots. The algorithm is shown in Figure 1.

3 The system architecture

We are developing a system to draw a feathered graph on Google Maps. The system architecture is shown in Figure 2.

The data gatherer will gather data from Web and analyze them automatically, but it is a future work.

The basic idea is that the feathered graph generator generates a png picture that suits the input data, and we use it as a marker for Google Maps. We make a half transparent image so that we can see the map through the feathered graph.

The feathered graph generator generates a feathered graph using RMagick [2]. RMagick is an extended library for using ImageMagick from Ruby. ImageMagick[3] is a set of software for manipulation and displaying images.

[2]http://www.simplesystems.org/RMagick/doc/index.html

[3]http://www.imagemagick.org/script/index.php

4 Trials for generating feathered graphs

4.1 Describing the used functions

In this section, we show 2×2 kinds of feathered graphs generated by combining the following functions.

(A) a cosine function and

(B) a linearly-decreasing function

for the function $F(point, value, d)$ and

(a) a single color gradation and

(b) a visible part of spectrum

for the function $G(x)$.

A cosine function for $F(point, value, d)$ decides values of around a source point according to the cosine curve. The relation of a value and a distance from the source point is shown at Figure 3. $distance$ is the constant given by a user, in which a source point has an effect. By this function, the area affected by a source point is constant. The sizes of feathered circles are the same. They are independent from values of source points.

A linearly-decreasing function for $F(point, value, d)$ decides values of around a source point according to a line of which slope is $max\_value/distance$, where max_value is the maximum value of source points and $distance$ has the same meaning in a cosine function (Figure 4). By this way, The size of a feathered circle depends on the value of a source point. The bigger value makes the bigger circle.

A single color gradation for $G(x)$ assigns a simple color to values. The example in this paper, we use blue. In the case of max_value, the color of the dot is blue which value is maximum, 255. In the case of the value is 0, the value of blue is 0 with the full transparency, that is no color.

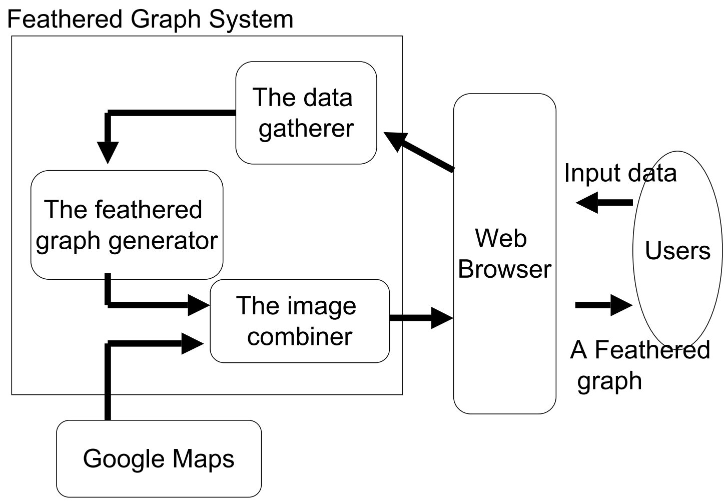

Figure 2. The system architecture.

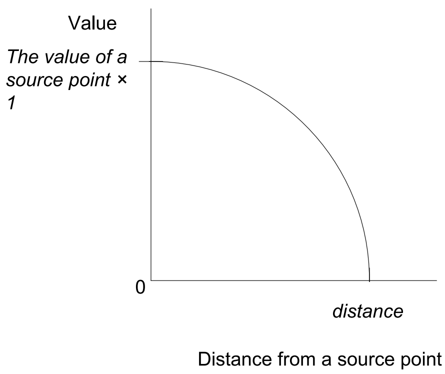

Figure 3. A cosine function.

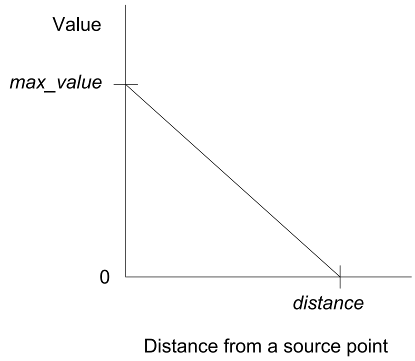

Figure 4. A linear-decreasing function.

A visible part of spectrum for $G(x)$ assigns a color according to the wavelength of spectrum. That is, in the case of max_value, the color is red. As decreasing the value, the color changes orange, yellow, light green, green, light blue, blue, and purple. In the case of the value is 0, the color is black with the full of transparency, that is no color.

4.2 Input data

The input data for our trials is the number of tourists in Kyoto, Japan. Kyoto is the most famous sightseeing area in Japan. There are many temples. Kyoto city publishes a report of the number of tourists at sightseeing spots every year. We get the data for trials from the report of 2007[4].

Table 2 shows the top 25 sightseeing spots. Multiple answers are allowed so that the total of percentage goes over 100.

4.3 Generated graphs

We show 4 kinds of feathered graphs of which input data is described in Subsection 4.2. In the all cases, $distance$ is 100 dots. We show default markers of Google Maps with ID shown in Table 2 on each sightseeing spots to recognize their locations easily.

1. The graph with a cosine function and a single color is shown in Figure 5.

[4] It can find at http://raku.city.kyoto.jp/kanko_top/kanko_chosa.html (in Japanese).

Table 2. The top 25 sightseeing spots in Kyoto (2007).

ID	Name of sightseeing spot	number of tourists (percentage)
A	Kiyomizu-dera	21.2
B	Arashiyama	15.9
C	Kinkaku-ji	12.0
D	Ginkaku-ji	9.7
E	Nanzen-ji	9.5
F	Higashiyama Kodai-ji	7.2
G	Yasaka-jinjya	7.0
H	Nijou-jou	6.7
I	Sagano	6.4
J	Kurama, Kibune	6.1
K	Oohara	5.6
L	Touji	5.6
M	Shijo Kawaramachi	5.3
N	Heian-jingu	5.3
O	Kyoto Station	4.7
P	Shimogamo-jinja	4.7
Q	Kyoto Municipal Museum of Art	4.5
R	Chion-in	4.5
S	Sanjusangen-do	4.2
T	Kyoto Gosho	3.6
U	Nishi Honganji	3.6
V	Higashi Honganji	3.3
W	Ryouan-ji	3.1
X	Nishiki Ichiba	2.8
Y	Fushimi Inari Taisha	2.5

2. The graph with a linear-decreasing function and a single color is shown in Figure 6.

3. The graph with a cosine function and a spectrum color is shown in Figure 7.

4. The graph with a linear-decreasing function and a spectrum color is shown in Figure 8.

4.4 Discussions

The number of tourists are suitable for feathered graphs since it can be considered a local maximum variables. In the example shown in this paper, the values of 25 sightseeing spots are used. Some spots are close to each other, and their feathered circles overlapped. Especially, in a liner-decreasing function, the feathered circles for Higashiyama Kodai-ji (ID: F), Yasaka-jinjya (ID: G), Chion-in (ID: R) and Sanjusangen-do (ID: S) are included in the feathered circle of Kiyomizu-dera (ID: A) because their values are too small for the value of Kiyomizu-dera.

The eyes of human are not so sensitive about differences of colors. Therefore, in the feathered graphs shown in this paper, it seems harder to us to find differences of values in the single colored feathered graph than the spectrum feathered graph. A single colored feathered graph may be suitable for the values of source points are similar.

5 Conclusion

We propose a feathered graph in this paper, and show its drawing algorithm. We also show some images of a feathered graph on Google Maps. The features of the feathered graph are:

- We can easily understand data with geographical information by feathered graphs on maps.
- The functions $F(point, value, d)$ and $G(x)$ are defined freely by users.
- The algorithm can be parallelized easily so that the feathered graphs can visualize a huge amount of data as well as a small amount of data.

References

[1] P. M. Austin. Relation between measured radar reflectivity and surface rainfall. *Monthly Weather Review*, 115(5):1053–1071, May 1987.

[2] M.-P. Kwan. Gis methods in time-geographic research: geocomputation and geovisualization of human activity patterns. *Geografiska Annaler Series B Human Geography*, 86(4):267–280, 2004.

[3] A. M. MacEachren and M.-J. Kraak. Research challenges in geovisualization. *Cartography and Geographic Information Science*, 28(1):3–12, 2001.

[4] M. A. Oliver and R. Webster. Kriging: a method of interpolation for geographical information systems. *International Journal of Geographical Information Science*, 4(3):313–332, 1990.

[5] H. Wackernagel. *Multivariate geostatistics: an introduction with applications*. Springer, 2003.

[6] J. W. Wilson and E. A. Brandes. Radar measurement of rainfall-a summary. *Bulletin of the American Meteorological Society*, 60(9):1048–1058, 1979.

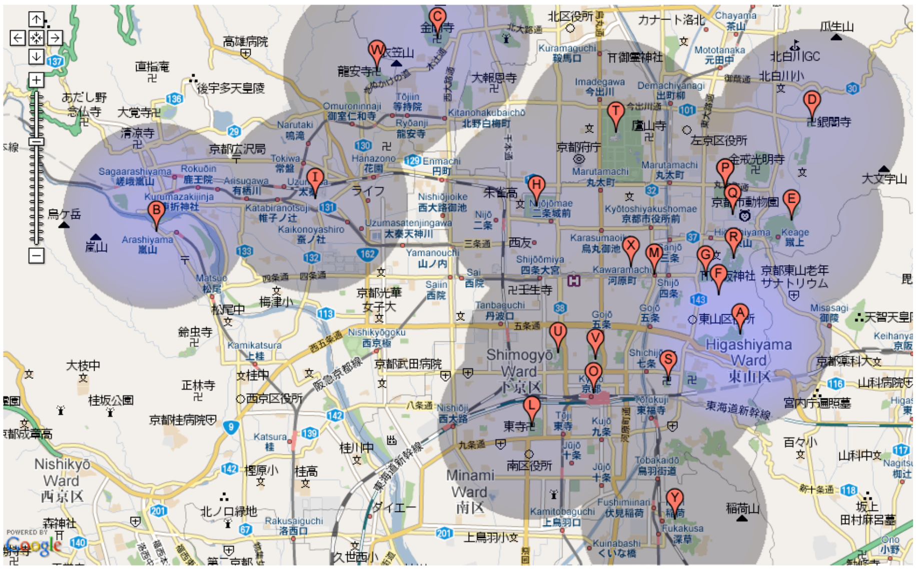

Figure 5. The graph with a cosine function and a single color.

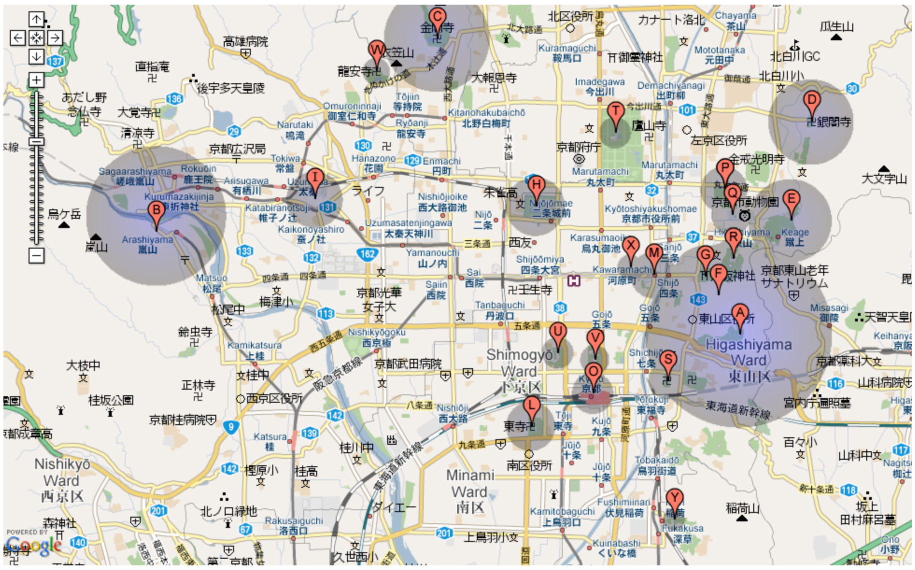

Figure 6. The graph with a linear-decreasing function and a single color.

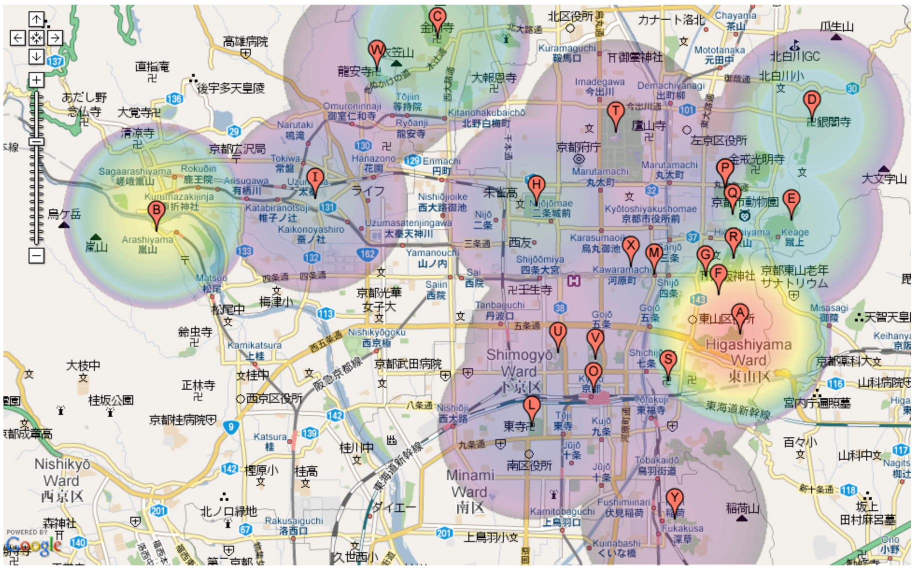

Figure 7. The graph with a cosine function and a spectrum color.

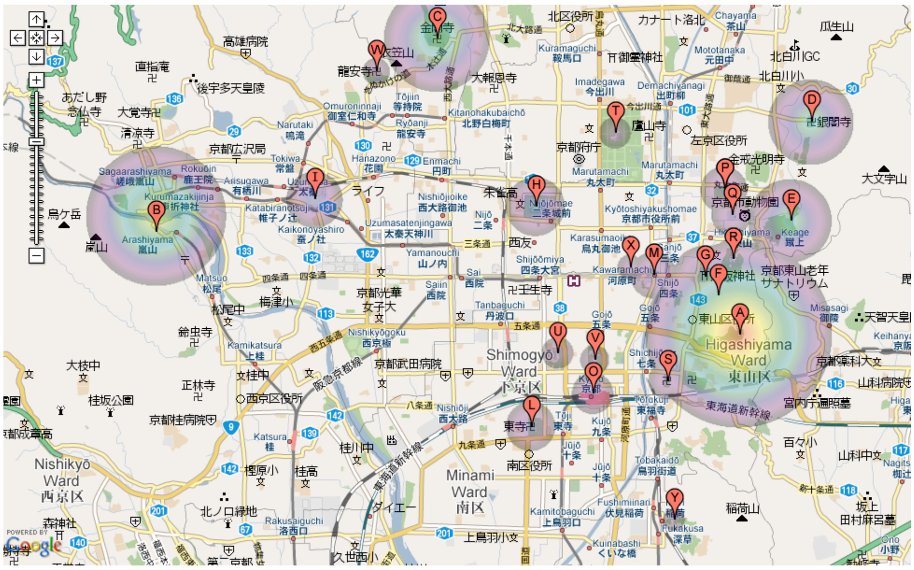

Figure 8. The graph with a linear-decreasing function and a spectrum color.