Vol.2017-DBS-165 No.5
Vol.2017-IFAT-128 No.5
2017/9/18

Design of Multiple Modified Features Based on a Map Analysis of Geographical Information

Kentaro Narikawa[1,a)] Daisuke Kitayama[2,b)] Kazutoshi Sumiya[1,c)]

Abstract: This paper presents a method for searching for modified maps that are similar to a user's query. Although users have many different reasons for using maps, there is a method of similarity search that can support user decision-making. Modified maps are simplified maps that are changed in some way. They are useful in terms of reflecting a user's specific purposes; however, there is no existing method for searching modified maps. In this paper, we propose a system that suggests a list of similar modified maps by analyzing the features of a modified map that the user inputs.

Keywords: Geographical information, Modified maps, Multiple features, Searching method

1. Introduction

These days, a large amount of content is available on the Web, and we can find what we want easily by searching with a query. As the Web has developed, a large amount of geographical information system has been released, for example, Google maps [1], Yahoo! maps [2], and Google Street View [3]. The expressions of geographical data are diverse, for example, there are three-dimensional maps, modified maps, bird's-eye maps, and aerial photographs. Such maps are generally called online maps. However, pre-existing online maps have some disadvantages. Most online maps show locations that a user does not need to know, which makes them cluttered and hard to read. In addition, it is difficult for users to customize pre-existing online maps for personal use because it is not possible sort out location visibility. Therefore, we focus on modified maps (Figure 1). Modified maps are a type of map that emphasizes and omits locations on a normal map according to a user ' s specific purpose. In addition, it is easy for us to read modified maps visually, as they express only the necessary information. Modified maps that reflect different purposes for users currently exist.

Moreover, there are many kinds of modified maps. For example, maps that show a route to a destination or location relationships, include illustrations, and show shortest paths. As a result, there are various types of modified maps.

Hence, in this research, our ultimate aim is to automatically generate modified maps that reflect a user's intentions. Such a service, called Destination Maps, has been implemented. It was a part of Bing Maps [4]and generated specific modified maps au-

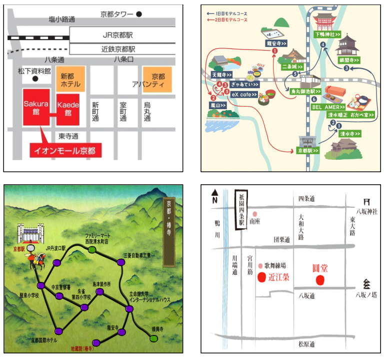


Fig. 1 Examples of modified maps

tomatically. However, the map generation only omitted locations and roads based on general importance values and it did not effectively emphasize specific information, which is a feature of modified maps. In this paper, we focus on an initial step before the automatic generation of modified maps, which is to search for similar modified maps. We need a method to obtain similar modified maps because similar previously modified maps enable the system to specify a user's interests.

In this paper, we present a method for searching for modified maps that are similar to a user's input map as a preprocessing phase for automatic modified map generation. In Section 2, we review related work. In Section 3, we present an overview of the search system for similar modified maps, explain the search method, and describe two databases used for this system. In Section 4, we describe the algorithm of the search system in detail, and in Section 5, describe the experiments, including the methods, requirements, results, and discussions. Finally, in Section 6, we state the conclusions and future work.

2. Related Work

There are many studies related to geographical information, especially regarding maps. The fundamental research on maps is Egenhofer's research, which analyzes the topological relation-

1 Kwansei Gakuin University, 2-1 Gakuen, Sanda, Hyogo 669–1337, Japan
2 Kogakuin University, 1-24-2 Nishi-shinjuku, Shinjuku, Tokyo 163–8677, Japan
a) fll83686@kwansei.ac.jp
b) kitayama@cc.kogakuin.ac.jp
c) sumiya@kwansei.ac.jp

ships of various maps focusing on intersections [5]. Furthermore, Weakliam proposed the personalization module of CoMPASS, which is a system that does not require explicit input from the user, but monitors all the implicit actions of the user as they browse maps looking for specific spatial content in terms of features and locations of interest [6]. This research is about developing a module for personalizing maps, so our research is related with it in terms of personalization. Tezuka proposed a method of measuring the cognitive significances of place names and proposed a simple user-friendly interface that corresponds to the diverse intentions of users [7]. This research implements a way to read maps easily based on cognitive psychology, so our research is similar with it in terms of focusing on map features that are easy to read. In addition, Tezuka considered expanding the semantic area of location names to focus on the concept of metonymy and nominative particles connecting to location names on Web pages [8]. This research differs from our research in terms of generating modified maps in the present phase, but is thought to be similar in terms of how to treat extracted locations on modified maps.

There are also many studies related to modified maps. Kobayashi suggested a method for automatically generating a street view for the analysis of emphasized geographical objects and pre-existing modified maps [9]. In addition, Kopf presented a system for creating destination maps to provide users a way to navigate to a given location from anywhere in a given area of interest [10]. This research is related in terms of focusing on modified maps that navigate to some destinations based on a user's interest. Honda presented a prototype system for the automated generation of deformed maps based on a psychological model in the field of cognitive maps. They focus on road deformation and landmark relocation [11]. Casakin established a method to schematize real spaces using a branch model and taxonomy for making wayfinding maps [12]. Moreover, Kajita suggested a method of relocating landmarks accompanied with road modification that depends on the orthogonalization of famous streets with vector morphing [13]. These past studies about generating modified maps focus on landmarks related to location display. Personalization depending on relationships with locations of interest to a user are not considered. This research differs from our research in terms of not focusing on personalization using relationships with locations of interest to a user.

Furthermore, there is a study that does not generate modified maps but associates real space coordinates with pre-existing modified maps. Kitayama suggested a method of inferring where locations in modified maps are located in real space corresponding to the deformation of the modified maps [14]. Our research is an application of their research.

3. System Overview

The proposed system generates modified maps that can match a user's query. This system consists of two phases. The first phase is to search for modified maps in a database that are related to a user's input. The second step is to edit the chosen modified maps from the database according to the user's preferences.

Our research goal is to generate modified maps that match a user's preferences. However, in this paper, we explain the first

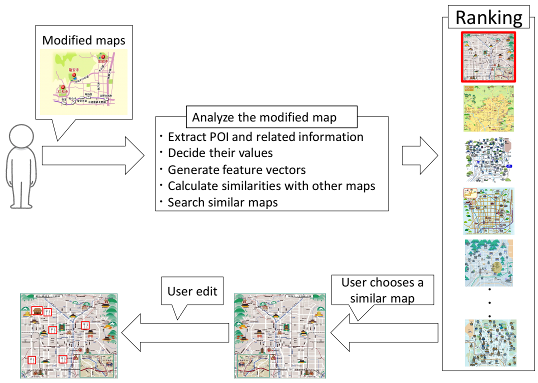


Fig. 2 System overview

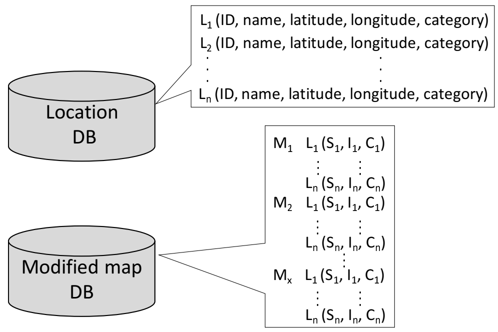


Fig. 3 Structure of the two databases

phase, which is how to search for modified maps that are related a user's input map in a modified map database and how to create the modified map database (Figure 2).

3.1 Modified Map Search

First, there are three possible kinds of user inputs in this research: keywords only, modified maps only, and a mixture of keywords and modified maps. However, in this paper, we explain the main case, which is modified map input only because it is the most primitive input.

We explain the simplified flow modified map search. This system extracts the POIs, which are locations in the modified maps used by the input, and weights the extracted POIs in terms of the kinds of features written in the modified maps. From the weighted POIs, this system creates a query vector and calculates the similarity of the modified maps to many other modified maps in the database.

3.2 Databases

Second, we describe two databases. In this research, we need to prepare two databases: a location database and a modified map database (Figure 3). In Figure 3, L_n is a specific location and M is a specific modified map. In addition, in the modified map database image, S is the text size feature, and I is the illustration feature, and C is the text color feature.

3.2.1 Location Database

In this section, we describe the location database, which is needed by the modified map database to determine a location. Its category, latitude, and longitude, is shown in the modified maps. In short, the location database must include five pieces of information: location ID, location name, latitude, longitude, and category in order to determine what kinds of locations are shown in the modified maps(Table1).

Then, we used the Foursquare search application programming interface (API) for creating the location database. Foursquare [15] is a social networking website based on geographical information. Their Web API is published freely, and a many web services have been developed using the Foursquare API.

Table 1 Example of Location DB

Locations ID	Latitude	Longitude	Locations name	category
4b74f523f964a52058f92de3	35.02194364	135.7201689	Myoshinji Temple	Buddhist Temple
4cff4b7c6923721efea7be3b	35.01974332	135.7187724	Taketori	Japanese Restaurant
4f1bceb30039e5a96263f120	35.01975204	135.7203421	Myoshinji bus stop	Bus Stop
4fdc3c74e4b0767cc6b65c1a	35.01826842	135.7192672	SMRI	Miscellaneous Shop
52970cf811d20a5e64b9d0bf	35.019471	135.719545	Owariya	Diner
4c4a568142b4d13a2cd80d7d	35.03425158	135.7185088	Ryoan-ji Rock Garden	Garden

3.2.2 Modified Map Database

In this section, we present the modified map database, which includes displayed locations, text size information, illustration information emphasis, and text color emphasis. This system calculates the similarity of both the input and output candidates of the modified maps in this database using these multiple features.

The modified map database is composed of vast quantities of modified map images from web pages. The stored modified maps are accompanied by metadata indicating what locations are shown and how they are emphasized.

The features of almost all modified maps consist of location names, their text size, illustrations, and the difference in text colors. We focus on these dimensions to define the features of modified maps. To consider user interests from the input modified maps, we focus on specifying which locations exist and how they are emphasized in the modified maps. In addition, we also focus on each text color and which location illustrations exist in the modified maps. We think these dimensions reflect a user's interest deeply. Hence, we have to extract these features to determine the overall feature of the modified maps. To determine which texts in the modified maps are locations or not, we rely on a location dictionary. In this research, we use the Foursquare API as the location dictionary.

We define all location names in input modified maps as a point of interest (POI). A POI is a location that a user is interested in. In this research, a POI is an extracted location name from input modified maps. To extract the POIs, we need to determine how to obtain the location names in the modified maps. The solution is to use optical character recognition (OCR). OCR is the recognition of printed or written text characters by a computer. This method is also able to analyze scanned images.

First, we state how to extract the text and text size of location names. Almost all modified maps have some text, however, it is not always written as text information. In short, some text is often written as image information. Hence, text information from input modified maps must be extracted using OCR, which can obtain the text size.

Second, we state how to extract illustrations that show locations. OCR can determine what is text and where it is in modified map, so that the system can find illustrations that indicate locations using illustration cognition technology, which uses Google image search and applies it around the text locations.

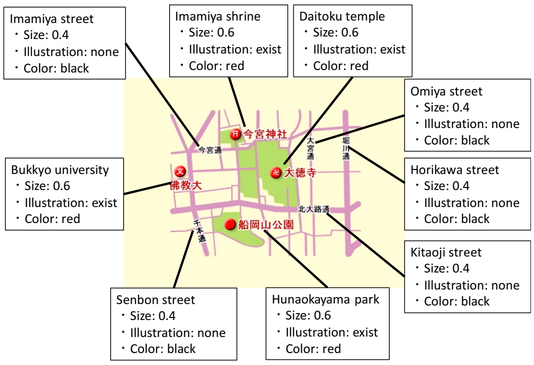


Fig. 4 Example of extracted POI and related information

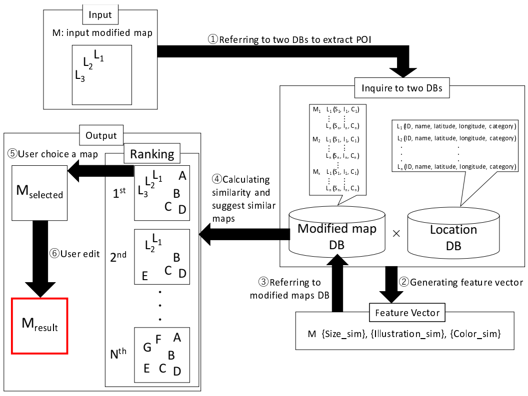


Fig. 5 Algorithm flow

Third, we extract the text color of the location name. If the background, which is behind the text, is a simple color, the system can recognize the text color because it is the least frequent RGB value in the area that is surrounded by the MBR that includes the text. In contrast, if the background uses many colors, the system can recognize the text color because it is the most frequent RGB value in the text MBR.

These three basic features are important for the automatic extraction of modified map features (Figure 4).

4. Analyzing Geographical Information

There are three kinds of user inputs, however, we suggest using modified maps, which are the most primitive type of input. There are five steps for generating a ranking for similar modified maps. In Section 4.1, we explain the method to extract text information and size from modified map user input. In Section 4.2, we explain how to extract the POIs that users are interested in for some locations from the modified map user input, and we describe the weighting of the extracted POIs. Next, in Section 4.3, we introduce how to generate the feature vectors from the POIs, and then explain how to rank using cosine similarity. We show the flow of this algorithm in Figure 5, where A to F are locations related to POIs.

4.1 User Input

There are three kinds of user input in this system: keywords,

Table 2 Example of generated feature vectors

map_id	location	size_deviation	illust_appearance_ratio	illust_existence	color_saliency	illust_rate	color_r	color_g	color_b
143	Nijo castle	0.16875	0.333333333	0	0.142857143	0.208333	255	255	255
143	Kyoto Tower	-0.13125	1	1	0.111111111	0.541667	0	0	0
143	Kyoto Imperial Palace	-0.13125	1	1	0.111111111	0.541667	0	0	0
143	Kyoto Aquarium	-0.13125	1	1	0.111111111	0.541667	0	0	0
143	Kyoto station	0.16875	0	0	0.142857143	0	255	255	255
143	Suburb of Kyoto station	0.16875	0	0	0.142857143	0	255	255	255
143	Uzumasa Movie Village	-0.13125	1	1	0.111111111	0.541667	0	0	0
143	Arashiyama・Saga	0.16875	0	0	0.142857143	0	255	255	255
143	Higashiyama・Gion	0.16875	0	0	0.142857143	0	255	255	255
143	Katsura River	-0.13125	0.666666667	1	0.111111111	0.395833	0	0	0
143	Kawaramachi	0.16875	0	0	0.142857143	0	255	255	255

modified maps prepared by the user, and both keywords and modified maps. However, we describe the modified map input only because modified maps for a natural query and are a primitive form for finding and ranking similar modified maps.

4.2 Generating Feature Vectors

POIs have three features: the text size of the location names, the proportion of illustrations used to display the locations, and the text color of the location names.

The value of the location name text size is determined by the absolute value of the deviation of the text size of all the location names in the input modified map. The proportion of illustrations is calculated using the number of illustrations divided by the number of all location names in the modified maps. If there are none, the value is 0. The value of the text color of the location names is determined using the inverse number of locations that use emphasized color in the modified map. These values are used to determine the feature vectors of each modified map.

As we mentioned above, we must generate feature vectors to find similar modified maps. These three features are treated as vectors. The structure of these feature vectors is shown in Table 2.

4.3 Calculating Similarities

Each modified map has three features: text size, illustration proportion, and text color. This algorithm calculates the similarity of the modified map feature vectors using cosine similarity or euclidean distance.

$$Sim(x,y) = \frac{\sum_{i=1}^{|V|} x_i \cdot y_i}{\sqrt{\sum_{i=1}^{|V|}(x_i)^2} \cdot \sqrt{\sum_{i=1}^{|V|}(y_i)^2}} \tag{1}$$

$$Euclidean\_distance = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{2}$$

Here, x and y are the feature vectors of each map and V is the number of all locations (1)(2).

From the input modified maps, feature vectors can be extracted. In addition, this system compares each feature vector of the input modified map to all pre-existing modified maps in the modified map database. To determine similar modified maps, this system combines the similarity values of each feature vector of a modified map. To combine the similarity values of each feature vector, we use a weighted linear combination as follows.

$$\begin{aligned} Com\_Sim(M_i, M_j) &= \alpha * Sim(M_{i\_size}, M_{j\_size}) \\ &+ \beta * Sim(M_{i\_illust}, M_{j\_illust}) \\ &+ \gamma * Sim(M_{i\_color}, M_{j\_color}) \end{aligned} \tag{3}$$

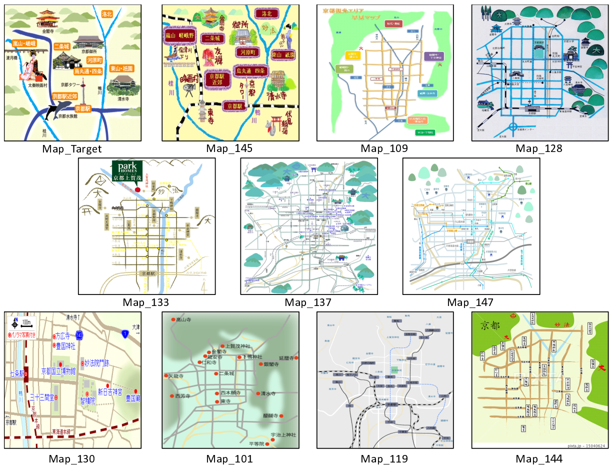


Fig. 6 Modified maps which used in the experiment

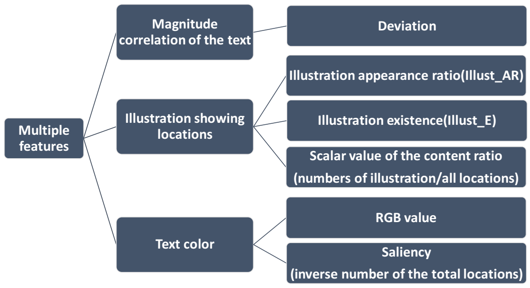


Fig. 7 Multiple features verified in this experiment

Here, Com_Sim means combined similarity and α, β, and γ are weights (3).

Afterwards, this system sorts similar modified maps according to the highest combined similarity. Finally, the user can choose the modified map that he/she wants.

5. Experiment

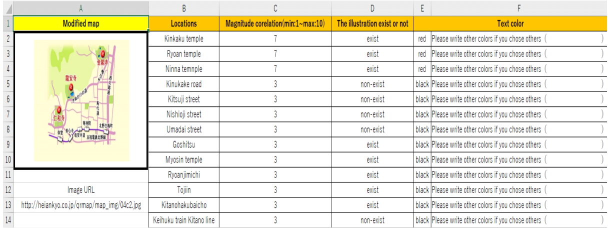
5.1 Method and Requirements

The purpose of this experiment is evaluating and finding which combination of features can present most similar rankings of modified maps. Figure 6 is the modified maps we used in this experiment.

We performed experiments to verify the ranking of similar maps using multiple features. In this experiment, we focused on three features of modified maps: the magnitude correlation of the texts, the illustrations in the modified maps, and the color of the texts. The illustration features have three more dimensions and text color features have two more dimensions internally. We show the multiple features used in this experiment in detail in Figure 7.

To calculate similarities between modified maps, this system need to calculate similarities with using these 6 features. The features of deviation, illustration appearance ratio, illustration existence, and saliency are calculated by cosine similarity. On the other hand, the features of scalar value of the content ratio and RGB value are calculated by euclidean distance as similarity. Furthermore, We use Jaccard index as one of the comparative features which is showing similarity between modified maps.

We determined seven rankings from this combination of mul-



	A	B	C	D	E	F
1	Modified map	Locations	Magnitude corelation(min:1~max:10)	The illustration exist or not	Text color	
2		Kinkaku temple	7	exist	red	Please write other colors if you chose others ( )
3		Ryoan temple	7	exist	red	Please write other colors if you chose others ( )
4		Ninna temnple	7	exist	red	Please write other colors if you chose others ( )
5		Kinukake road	3	non-exist	black	Please write other colors if you chose others ( )
6		Kitsuji street	3	non-exist	black	Please write other colors if you chose others ( )
7		Nishioji street	3	non-exist	black	Please write other colors if you chose others ( )
8		Umadai street	3	non-exist	black	Please write other colors if you chose others ( )
9		Goshitsu	3	exist	black	Please write other colors if you chose others ( )
10		Myosin temple	3	exist	black	Please write other colors if you chose others ( )
11		Ryoanjimichi	3	exist	black	Please write other colors if you chose others ( )
12	Image URL	Tojiin	3	exist	black	Please write other colors if you chose others ( )
13	http://heiankyo.co.jp/qrmap/map_img/04c2.jpg	Kitanohakubaicho	3	exist	black	Please write other colors if you chose others ( )
14		Keihuku train Kitano line	3	non-exist	black	Please write other colors if you chose others ( )

Fig. 8 Example of map that subjects evaluated

tiple features and confirmed that the similar modified maps were sorted appropriately. We used 11 provisionally modified maps about Kyoto that can be used as both input and output in this experiment. We regarded a combination of three dimensions as a feature vector, calculated each feature vector using cosine similarity and euclidean distance, and calculated the similarity of the values using a weighted addition. We present the final results in the order of descending similarity. For the weighted addition, we used a linear combination to add different vectors.

We created a location database and modified map database using real data and confirmed the proposed algorithm using these two databases. In addition, we added an algorithm to calculate similarity for multiple features and examined the output rankings for the similar modified maps produced by the different combinations.

In this experiment, we focused on Kyoto, which is a typical place in Japan. The location database was created by the Foursquare venue search API and we obtained 57,449 data locations consisting of location IDs, names, latitude, longitude, and category in this experiment. The category is given for each location on the Foursquare map.

In addition, we used Crowd Works [16], which is a crowd sourcing site to collect the average data of each vector and eliminate bias. We obtained data from 960 modified maps because we asked subjects to answer three questions per modified map. We show an example of the questions that the subjects answered in Figure 8.

We also created a modified map database by collecting 320 modified maps for Kyoto from web pages for this experiment. We prepared three different vectors that emphasize different features. All of the maps have different combinations of feature values. For the magnitude correlation vector, we adopted the degree of text size deviation. For the illustration vector, we adopted a method that yields 0 or 1 depending on the existence of the locations and that calculate illustration appearance ratio as average numbers of three subjects who answer the illustration exist, scalar value that indicates the ratio of location illustrations on the modified map. The scalar value of the ratio of illustrations is the number of illustrations divided by the number of all locations on the modified map. For the color vector, we adopted RGB values and the inverse number of the total locations as saliency.

In this experiment, as the final step for creating the feature vectors of each modified map, each vector was given weights. In short, to extract the features for each modified map, we adopted a linear weighted sum. However, in this step, we do not know what weights are appropriate, so we set all weights to the same value. More specifically, the weights are all one.

We asked 7 subjects to arrange modified maps in the order of descending similarity, and defined scores on each ranking in the way of the first is 10, the second is 9, and the 10th is 1. We regarded the ranking gathered in the order of descending scores as ideal or right ranking. After we defined right ranking, we evaluated which ranking made from each combinations of features is close to right data by using Spearman's rank correlation coefficient (4), where the difference between the value of each pair is d, and the number of dimensions is n.

$$r_s = 1 - \frac{6\sum_{i=1}^{n} d_i{}^2}{n(n^2 - 1)} \quad (4)$$

5.2 Result

Spearman's correlation coefficient of each feature are shown in Figure 9. As previous phase, we calculated correlation coefficient of each features(Table 3). Moreover, we can see the large difference of rankings made from each combinations(Table 4). In the results, each value was normalized from 0 to 1 using min-max normalization except Illust_rate similarity. The modified maps used in this experiment are Figure 6. Map_target is an input map, and the others are output maps that were compared to the input map.

Ideal	
Jaccard index	
Size_sim Illust_AR_sim Color_euclid_sim	→ A
Size_sim Illust_AR_sim Color_saliency_sim	→ B
Size_sim Illust_E_sim Color_euclid_sim	→ C
Size_sim Illust_E_sim Color_saliency_sim	→ D
Size_sim Illust_rate_sim Color_euclid_sim	→ E
Size_sim Illust_rate_sim Color_saliency_sim	→ F

Illust_appearance_ratio → Illust_AR

Illust_existence → Illust_E

	Jaccard	A	B	C	D	E	F
Spearman's correlation coefficient	0.685	0.540	0.491	0.467	0.406	0.019	0.019

Fig. 9 Spearman's correlation coefficient

Table 3 Correlation coefficient of each features

Combination	Correlation coefficient
size deviation and illust appearance ratio	0.241220644
size deviation and illust existence	0.196387493
size deviation and saliency	0.072898677
illust appearance ratio and saliency	-0.116162063
illust existence and saliency	-0.121672766

Table 4 Rankings

Ranking	Right	Jaccard	A	B	C	D	E	F
1	145	145	145	145	145	145	147	147
2	128	109	128	128	128	128	119	119
3	109	128	137	137	137	137	144	145
4	133	133	147	147	147	147	145	144
5	144	130	133	119	119	119	128	128
6	137	137	119	144	144	109	133	130
7	147	147	144	109	133	130	130	133
8	101	101	130	130	130	144	109	109
9	119	119	109	133	109	101	101	137
10	130	144	101	101	101	133	137	101

Table 5 Result

Combination	Size_sim	Illust_AR_sim	Color_euclid_sim	Weighted addition	Ranking	
Map_target-145	0.251	0.391	0.054	0.695	1	145
Map_target-109	0.035	0.000	0.063	0.098	2	128
Map_target-128	0.094	0.343	0.054	0.491	3	137
Map_target-133	0.035	0.132	0.102	0.269	4	147
Map_target-130	0.060	0.000	0.055	0.115	5	133
Map_target-137	-0.013	0.354	0.092	0.432	6	119
Map_target-147	0.109	0.218	0.081	0.408	7	144
Map_target-101	0.000	0.000	0.054	0.054	8	130
Map_target-119	0.193	0.000	0.070	0.263	9	109
Map_target-144	-0.053	0.240	0.054	0.242	10	101

Combination	Size_sim	Illust_E_sim	Color_saliency_sim	Weighted addition	Ranking	
Map_target-145	0.251	0.333	0.449	1.033	1	145
Map_target-109	0.035	0.000	0.291	0.326	2	128
Map_target-128	0.094	0.302	0.224	0.620	3	137
Map_target-133	0.035	0.000	0.089	0.125	4	147
Map_target-130	0.060	0.000	0.250	0.310	5	119
Map_target-137	-0.013	0.333	0.292	0.612	6	109
Map_target-147	0.109	0.167	0.115	0.390	7	130
Map_target-101	0.000	0.000	0.176	0.176	8	144
Map_target-119	0.193	0.000	0.194	0.387	9	101
Map_target-144	-0.053	0.149	0.157	0.254	10	133

Combination	Size_sim	Illust_AR_sim	Color_saliency_sim	Weighted addition	Ranking	
Map_target-145	0.251	0.391	0.449	1.090	1	145
Map_target-109	0.035	0.000	0.291	0.326	2	128
Map_target-128	0.094	0.343	0.224	0.661	3	137
Map_target-133	0.035	0.132	0.089	0.257	4	147
Map_target-130	0.060	0.000	0.250	0.310	5	119
Map_target-137	-0.013	0.354	0.292	0.632	6	144
Map_target-147	0.109	0.218	0.115	0.441	7	109
Map_target-101	0.000	0.000	0.176	0.176	8	130
Map_target-119	0.193	0.000	0.194	0.387	9	133
Map_target-144	-0.053	0.240	0.157	0.345	10	101

Combination	Size_sim	Illust_rate_sim	Color_euclid_sim	Weighted addition	Ranking	
Map_target-145	0.251	1.742	0.054	2.046	1	147
Map_target-109	0.035	1.553	0.063	1.651	2	119
Map_target-128	0.094	1.814	0.054	1.961	3	144
Map_target-133	0.035	1.794	0.102	1.931	4	145
Map_target-130	0.060	1.729	0.055	1.844	5	128
Map_target-137	-0.013	1.475	0.092	1.554	6	133
Map_target-147	0.109	3.334	0.081	3.524	7	130
Map_target-101	0.000	1.553	0.054	1.607	8	109
Map_target-119	0.193	2.664	0.070	2.926	9	101
Map_target-144	-0.053	2.053	0.054	2.055	10	137

Combination	Size_sim	Illust_E_sim	Color_euclid_sim	Weighted addition	Ranking	
Map_target-145	0.251	0.333	0.054	0.638	1	145
Map_target-109	0.035	0.000	0.063	0.098	2	128
Map_target-128	0.094	0.302	0.054	0.449	3	137
Map_target-133	0.035	0.000	0.102	0.137	4	147
Map_target-130	0.060	0.000	0.055	0.115	5	119
Map_target-137	-0.013	0.333	0.092	0.412	6	144
Map_target-147	0.109	0.167	0.081	0.356	7	133
Map_target-101	0.000	0.000	0.054	0.054	8	130
Map_target-119	0.193	0.000	0.070	0.263	9	109
Map_target-144	-0.053	0.149	0.054	0.150	10	101

Combination	Size_sim	Illust_rate_sim	Color_saliency_sim	Weighted addition	Ranking	
Map_target-145	0.251	1.742	0.449	2.441	1	147
Map_target-109	0.035	1.553	0.291	1.878	2	119
Map_target-128	0.094	1.814	0.224	2.132	3	145
Map_target-133	0.035	1.794	0.089	1.918	4	144
Map_target-130	0.060	1.729	0.250	2.039	5	128
Map_target-137	-0.013	1.475	0.292	1.753	6	130
Map_target-147	0.109	3.334	0.115	3.558	7	133
Map_target-101	0.000	1.553	0.176	1.728	8	109
Map_target-119	0.193	2.664	0.194	3.051	9	137
Map_target-144	-0.053	2.053	0.157	2.158	10	101

5.3 Discussion

As a result, method of Jaccard index is better than other methods. Next, the combination of deviation, illust appearance ratio and color euclidean distance is higher. We found that illust appearance ratio and color can affect similar maps. Simultaneously, we can say the requirement of similar maps is depending on the number of same locations in modified maps.

However in this experiment, right data was made by a few subjects, so we have to observe scattering of subjects' answers by using fleiss's kappa index. If the scattering is small among subjects, we can say right data of this experiment is almost true. In contrast, if the scattering is big, more number of subjects are needed because this result may be accidentally happened. Furthermore, we observe how does rankings change bu calculating not weighted addition but multiplication of each feature vectors, and we try other way of evaluate like nDCG. In addition, in terms of discussions, we analyze scattering of both higher rank and lower rank. Not only quantitative evaluation but also qualitative evaluation is needed by noticing the semantic difference of rankings.

For the features of the user's chosen modified maps, the extent of emphases were consistent. It is difficult to find the difference and determine which features are effective for similar modified maps. We are sure that we can perform an experiment that uses input modified maps with different features with different amounts of emphasis.

Finally, with respect to the illustration vector, we do not know which dimensions are effective because the order of rankings made from the existence of illustrations and proportion of illustrations features are the same. Hence, we should modify the dataset accordingly. However, we have to be careful to add to the dataset modified maps in which the proportion of illustrations is lower and that have many different illustrations compared with the input modified map. In short, we need to add experimental data for which the proportion of illustrations is similar but the number and kind of illustrations are not similar.

6. Conclusions

In this paper, using arbitrary input maps modified by users, we proposed a search method for finding similar modified maps. We created location and modified map databases. We calculated the feature values of the modified map with respect to the magnitude correlation of the texts, existence of illustrations, and texts colors in the modified map, and conducted an experiment that ranks of similar maps in descending order after each similarity is determined.

In future work, we need to perform another experiment that addresses the issues found in this experiment and obtain better rankings of more similar modified maps. Next, to evaluate the proposed algorithm, we should prepare comparative algorithms and determine the most appropriate algorithm from an analysis and discussion of the differences. In addition, in this paper, we investigate only the modified map input, but we simultaneously need to designed a system that can also treat keyword input. Moreover, we presented a method for expressing similar modified maps, and did not present a method for generating or editing modified maps in order to match user intentions. This is an issue we need to consider deeply.

7. Acknowledgement

This work was supported by JSPS KAKENHI Grant Numbers 16K12536 and 16H01722.

References

[1] Google Maps, available from ⟨https://www.google.co.jp/maps⟩.
[2] Yahoo! Maps, available from ⟨http://map.yahoo.co.jp/⟩.
[3] Google Street View, available from ⟨https://www.google.com/streetview/⟩.
[4] Bing Maps, available from ⟨https://www.bing.com/maps/⟩.
[5] J.M. Egenhofer.. A Model for Detailed Binary Topological Relationships.Geomatica. 1993, vol.47, no. 3, p. 261-273.
[6] W. Joe and B. Michela and W. David. Implicit Interaction Profiling for Recommending Spatial Content. In Proceedings of the 13th annual ACM international workshop on Geographic information systems. 2005, p. 285-294.
[7] T. Tezuka and Y. Yokota and M. Iwaihara and K. Tanaka. Extraction of Cognitively-significant Place Names and Regions from Web-based Physical Proximity Co-occurrences. Web Information Systems–WISE 2004. 2004, p. 113-124.
[8] T. Tezuka and R. Lee and Y. Kambayashi and H. Takakura. Web-based Inference Rules for Processing Conceptual Geographical Relationships. Web Information Systems Engineering, 2001. In Proceedings of the Second International Conference. IEEE, 2001, vol.2, p. 14-21.
[9] K. Kobayashi and D. Kitayama and K. Sumiya. Cinematic street: Automatic Street View Walk-through System Using Characteristics of Modified Maps. Web and Wireless Geographical Information Systems. 2011, p. 142-158.
[10] J. Kopf and M. Agrawala and D. Bargeron and D. Salesin and M. Cohen. Automatic Generation of Destination Maps. ACM Transactions on Graphics (TOG). 2010, vol. 29, no. 6, p. 158.
[11] H. Honda and K. Yamamori and K. Kajita and J. Hasegawa. A System for Automated Generation of Deformed Maps. MVA. 1998, p. 149-153
[12] H. Casakin and T. Barkowsky and A. Klippel and C. Freksa. Schematic Maps as Wayfinding Aids. Spatial cognition II. 2000, p. 54-71.
[13] K. Kajita and K. Yamamori and J. Hasegawa. Development of Deformation Map Automatic Generation System. Information Processing Society of Japan Journal. 1996, vol. 37, no. 9, p. 1736-1744.
[14] D. Kitayama and K. Sumiya. A Deformation Analysis Method for Artificial Maps based on Geographical Accuracy and its Applications. In Proceedings of the 2nd Joint WICOW/AIRWeb Workshop on Web Quality. ACM. 2012, p. 19-26.
[15] Foursquare, available from ⟨https://ja.foursquare.com/⟩.
[16] Crowd Works, available from ⟨https://crowdworks.jp/⟩.