Regular Paper

Clustering and Visualizing Functionally Similar Regions in Large-Scale Spatial Networks

Takayasu Fushimi[1,a)] Kazumi Saito[2,b)] Tetsuo Ikeda[2,c)] Kazuhiro Kazama[3,d)]

Received: October 3, 2016, Accepted: March 3, 2017

Abstract: We address the problem of extracting functionally similar regions in urban streets and regard such regions as spatial networks. For this purpose, based on our previous algorithm called the FCE method that extracted functional clusters for each network, we propose a new method that efficiently deals with several large-scale networks by accelerating our previous algorithm using lazy evaluation and pivot pruning techniques. Then we present our new techniques for simultaneously comparing the extracted functional clusters of several networks and an effective way of visualizing these clusters by focusing on the fact that the maximum degree of the nodes in spatial networks is restricted to relatively small numbers. In our experiments using urban streets extracted from the OpenStreetMap data of four worldwide cities, we show that our proposed method achieved a reasonably high acceleration performance. Then we show that the functional clusters extracted by it are useful for understanding the properties of areas in a series of visualization results and empirically confirm that our results are substantially different from those obtained by representative centrality measures. These region characteristics will play important roles for developing and planning city promotion and travel tours as well as understanding and improving the usage of urban streets.

Keywords: k-medoids clustering, pivot pruning, functional cluster, Z-score, spatial network

1. Introduction

In such diverse fields as sociology, biology, physics, and computer science [17], studies of the structures and functions of large complex networks are attracting a great deal of attention. As a particular class, we focus on the spatial networks embedded in real spaces such as urban streets whose nodes occupy precise positions in two- or three-dimensional Euclidean space and whose links are real physical connections [3]. In this paper, we address the problem of clustering and visualizing functionally similar regions as functional clusters [9] by focusing on urban streets, which we regard as large spatial networks. Such regional characteristics will play important roles for developing and planning city promotion and travel tours as well as understanding and improving the usage of urban streets.

Compared with the conventional issue of extracting communities from networks [18], our issue shares the idea that the nodes in networks are divided into several groups. However, our issue is significantly different from the conventional one because we focus on the functional properties of nodes derived from a network structure [9]. For instance, for social networks where each node corresponds to a person, our objective is to extract groups of similar persons in terms of positions and/or roles with respect to others, such as the main members of each cluster, where these nodes (persons) are not necessarily connected directly to each other, instead of extracting the communities themselves that are typically defined as densely connected subnetworks. Note that such functional properties can be assumed in a wide variety of networks, but in this paper we focus on spatial networks constructed by mapping the ends and intersections of streets into nodes and the streets between the nodes into links. For these networks, node functionality is defined through a probability vector obtained from a random walk process [9]. We believe that extracting groups of functionally similar nodes in a spatial network is a critical research topic. Examples of such functional clusters might include parts of streets constructed in planned cities like lattices and those reflected by geographical restrictions like culde-sacs.

To extract functional clusters, we employ our previous algorithm called the Functional Cluster Extraction (FCE) method. The FCE method consists of two phases, the calculation of feature vectors (or functional vectors) through a random walk process, and the clustering of these vectors by the k-medoids method based on a greedy algorithm, where the latter clustering phase requires a huge computational cost. More specifically, let N be the number of functional vectors, which equals the number of nodes in the network, and let S be the dimension of the functional vectors, which equals the time steps of the random walk process. After calculating the pair-wise distance of these vectors with computational cost $O(N^2S)$, we run the k-medoids clustering phase with computational cost $O(KN^2)$ when we have enough memory space

[1] School of Computer Science, Tokyo University of Technology, Hachioji, Tokyo 192–0982, Japan
[2] School of Management and Information, University of Shizuoka, Shizuoka, Shizuoka 422–8526, Japan
[3] Faculty of Systems Engineering, Wakayama University, Wakayama, Wakayama 640–8510, Japan
[a)] takayasu.fushimi@gmali.com
[b)] k-saito@u-shizuoka-ken.ac.jp
[c)] t-ikeda@u-shizuoka-ken.ac.jp
[d)] kazama@ingrid.org

Table 1 Notation.

Symbol	Description and Definitions
V	Set of nodes
E	Set of links
$\Gamma(u)$	Set of adjacency nodes of node u
R	Set of representative nodes, $R \subset V$
P	Set of pivot nodes, $P \subset V$
N	Number of nodes, $N = \|V\|$
S	Dimension of functional vectors
K	Number of medoids (clusters), $K = \|R\|$
H	Number of pivots, $H = \|P\|$
$\mathbf{y}_s$	N-dimensional prob. vector of a random walk process at step s
$\mathbf{x}_u$	S-dimensional functional vector of node u
$\rho(u, v)$	Cosine similarity between functional vectors of node u and v
$\mu(u; R)$	Maximum similarity of node u, $\mu(u; R) = \max_{r \in R}\{\rho(u, r)\}$
$d(u, v)$	Euclidean distance between functional vectors of node u and v
$f(R)$	Objective function of k-medoids clustering
$g(w, R)$	Marginal gain of objective function
V_j	Set of nodes with degree j
$V^{(k)}$	Set of nodes belonging to functional cluster k
$V_j^{(k)}$	Set of nodes with degree j belonging to functional cluster k
$Z_j^{(k)}$	Z-score of degree j and functional cluster k

to store all of the $N(N-1)/2$ distances. However, when number N of the functional vectors is too large and the memory space is inadequate, we need to re-calculate most of the $N(N-1)/2$ distances for each of the K greedy steps at the k-medoids clustering phase, and thus the computational cost becomes $O(KN^2S)$. In our experiments below, the typical values for these variables are $K = 10$, $N = 100{,}000$, and $S = 10{,}000$, causing quite large computation times. To overcome this problem, we propose an accelerating clustering algorithm.

This paper is an extended version on our results that we previously presented: accelerating the k-medoids clustering phase of the FCE method using the lazy evaluation and pivot pruning techniques [8], and simultaneously comparing the extracted functional clusters of several networks and effectively visualizing them [7].

This paper is organized as follows. After explaining related work in Section 2, we describe the problem background of this paper in Section 3. In Section 4, we describe the details of the FCE method, our accelerating algorithm of k-medoids clustering, and the visualization method of extracted functional clusters. Then after explaining our datasets in Section 5, we evaluate the computational performance of our proposed algorithm and the characteristics of the extracted functional clusters in Section 6. Finally, we offer a conclusion in Section 7. For easy reference, we summarize the notation in **Table 1**.

2. Related Work

As mentioned above, the structures and the functions of large spatial networks have been studied by many researchers [2], [3], [15], [19], [21], [26]. From structural viewpoints, centrality measures have been widely used to analyze such networks [3], [21], especially by extending the conventional notions of centrality measures on simple networks into those of weighted networks [15], [19]. Traffic usage patterns in urban streets have been investigated from functional viewpoints [2], [26]. Unlike these previous studies, in this paper, as the intrinsic properties of these spatial networks, we extract functional clusters and naturally combine the structural and functional viewpoints in terms of such clusters.

Functional properties are assumed to exist in a wide variety of networks. Thus, in sociology, similar notions of node functions or roles have been studied as structural equivalence [14] and regular equivalence [6] with their extraction algorithms. These notions focus on local structures like relationships with adjacent nodes. Functional vectors in the FCE method, however, reflect not only local structures but also global ones through a random walk process.

Studies of community extraction are another main branch of complex network analysis. As mentioned above, our method extracts functional clusters [9]. This is because the representative methods for extracting communities as densely connected subnetworks, which include the Newman clustering method based on a modularity measure [18], the normalized cut method [24], or the ratio cut method [10] based on spectral graph analysis, cannot directly deal with such functional properties. Also, conventional notions of densely connected subnetworks such as k-core [23] and k-clique [20] will not function for this purpose. We naturally anticipate that these representative methods have an intrinsic limitation for extracting functional similar nodes. It might also be difficult to straightforwardly apply these conventional methods to spatial networks, because the maximum degree of nodes in each network is generally restricted to a relatively small number, since densely connected subnetworks are unlikely to appear in these networks.

In this paper, we focus on the FCE method that employs the k-medoids clustering method for dividing all nodes into groups of functionally similar nodes by the greedy maximization of the objective function. For clustering large-scale datasets, we can employ representative sampling algorithms like [1], [12]. However, since they compute approximated centers or clusters from stochastically selected, relatively small objects, the accuracy of the results is not guaranteed. Thus, we cite some existing methods that strictly solve the objective function and accelerate the algorithm using triangle inequality in the clustering fields. A well-known acceleration of the Lloyd algorithm called the Elkan algorithm [5] avoids redundant distance calculations in the K-means algorithm and divides N objects into K clusters. The acceleration results from the effective use of the lower and upper bounds on the distance and derives them from the triangle inequality. Unfortunately, this requires large amounts of memory space $O(NK)$ for storing the K lower bounds for each object. The Hamerly algorithm which extends the Elkan algorithm, employs only one lower bound for each object, resulting in the reduction of memory usage to $O(N)$ [11]. Recently, hybrid Elkan and Hamerly algorithms have been reported [4]. They treat the number of lower bounds as a variable parameter in the one to K range to best exploit the strength of each algorithm. By using pivots that efficiently select initial medoids and accelerate the convergence in iterative steps, Paterlini et al. proposed a fast algorithm of k-medoids clustering [22]. Unlike these existing methods, the FCE method based on a greedy approach theoretically guarantees a unique greedy solution with reasonably high quality because of the submodularity of the objective function.

3. Problem Description

In this paper, we extract functional clusters, each of which consists of functionally similar nodes from a given network. **Figure 1** shows an illustrative example of functional clusters extracted from a synthetic network that consists of five web-like graphs connected by a single link, where the clusters extracted by our FCE method are distinguished by red, green, and blue. These functional clusters are formed by nodes at the center part (red), the intermediate part (green), and the peripheral part (blue) for each web-like graph. Thus, for spatial networks constructed from urban streets, we expect to obtain such functional clusters as city centers with our designated resolution controlled by the number of clusters K. Here we emphasize that our method is potentially applicable to a wide range of complex networks including social networks constructed from the relationships of people and information networks constructed from paper citations. We expect to obtain functional clusters, such as groups of leaders from each community in social networks and the authors of outstanding papers from each field in information networks.

4. Methodology

In this section, we present the details of the FCE method and new techniques that 1) accelerate the k-medoids clustering phase of the FCE method; and 2) simultaneously compare the extracted functional clusters of several networks.

4.1 Functional Cluster Extraction Method

For extracting functional clusters [9], we revisit the FCE method that consists of two steps: calculation of functional vectors and the clustering of them.

Let $G = (V, E)$ be a given spatial network, where $V = \{u, v, w, \cdots\}$ and $E = \{(u, v), \cdots\} \subset V \times V$ stand for sets of nodes and links, where we denote the number of nodes by $N = |V|$. In this paper, we only consider undirected networks such that $(u, v) \in E$ implies $(v, u) \in E$, but our approach can be straightforwardly extended to deal with directional ones. For each node $u \in V$, we denote the set of its adjacent nodes by $\Gamma(u) = \{v \mid (u, v) \in E\}$. We define random walk probability $y_s(u)$ of node u at iteration step s by considering the following iterative process:

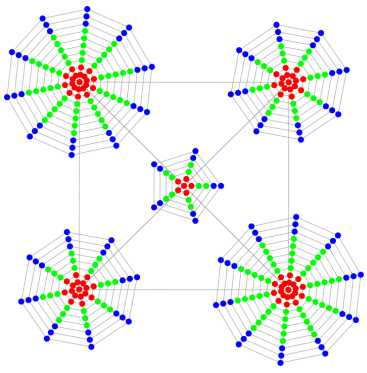


Fig. 1 Synthetic network that resembles an urban street and its functional clusters.

$$y_s(u) = \sum_{v \in \Gamma(u)} \frac{y_{s-1}(v)}{|\Gamma(v)|},$$

where $y_s(v) \geq 0$ and $\sum_{v \in V} y_s(v) = 1$. This model is a special version of PageRank where the teleportation jump probability α is set to 0. Note that under some mild conditions, $y_s(u)$ converges to a value proportional to the degree of node u, i.e., $|\Gamma(u)|/\sum_{v \in V}|\Gamma(v)|$. We focus on the PageRank score vectors at each iteration step s, i.e., $\{\mathbf{y}_0, \cdots, \mathbf{y}_S\}$, where we set the initial vector to $\mathbf{y}_0 = (1/N, \ldots, 1/N)$ and S stands for the final step of the iterations. Then for each node $u \in V$, we consider an S-dimensional vector defined by

$$\mathbf{x}_u = (y_1(u), \cdots, y_S(u)),$$

where $y_s(u)$ also corresponds to the PageRank score of node u at iteration step s. Here, $\mathbf{x}_u$ is called the functional vector of node u. The functional vector of each node contains not only local information like the degree of the node as a converged value but also global information accumulated through a random walk process like PageRank. Thus, by clustering the functional vectors, we can extract groups of similar nodes in terms of positions and/or roles with respect to the other nodes. Note that we set dimension S of the functional vector to a relatively large value, i.e., 10,000, because the diameters of the spatial networks in our experiments are generally large.

Based on the following cosine similarity, $\rho(u, v)$, between each pair of functional vectors, $\mathbf{x}_u$ and $\mathbf{x}_v$,

$$\rho(u, v) = \left\langle \frac{\mathbf{x}_u}{\|\mathbf{x}_u\|}, \frac{\mathbf{x}_v}{\|\mathbf{x}_v\|} \right\rangle,$$

we divide all the nodes into K groups of functional clusters by employing the k-medoids algorithm [25] due to its robustness. Formally, we maximize the following objective function with respect to the set of medoids $R \subset V$:

$$f(R) = \sum_{v \in V} \max_{r \in R} \rho(v, r).$$

To maximize objective function $f(R)$, we employ a greedy algorithm using the following marginal gain with respect to each candidate node w by establishing a set R of the already selected medoids:

$$\begin{aligned} g(w; R) &= f(R \cup \{w\}) - f(R) \\ &= \sum_{v \in V \setminus R} \max\{\rho(v, w) - \mu(v; R), 0\}, \end{aligned} \tag{1}$$

where $\mu(v\,; R) = \max_{r \in R}\{\rho(v, r)\}$ if $R \neq \emptyset$; otherwise $\mu(v\,; \emptyset) = 0$. Then we can summarize the greedy algorithm as follows:

(1) Initialize $k \leftarrow 1$ and $R_0 \leftarrow \emptyset$;
(2) Select $\hat{r}_k = \arg\max_{w \in V \setminus R_{k-1}} g(w; R_{k-1})$;
(3) Add $R_k \leftarrow R_{k-1} \cup \{\hat{r}_k\}$;
(4) If $k = K$, output $R_K = \{r_1, \cdots, r_K\}$.

From the obtained K medoids $R = \{r_1, \cdots, r_K\}$, we can calculate each functional cluster as

$$V^{(k)} = \left\{ v \in V; r_k = \arg\max_{r \in R} \{\rho(v, r)\} \right\}.$$

Based on the submodularity of the objective function, we are

guaranteed to obtain a unique greedy solution with reasonably high quality [16], unlike such other standard methods as K-means clustering. Moreover, for our problem setting, by setting $\mathbf{x}_v \leftarrow \mathbf{x}_v/\|\mathbf{x}_v\|$ for each node $v \in V$, we derive the following transformations:

$$g(w;\emptyset) = f(\{w\}) = \sum_{v\in V}\rho(v,w) = \left\langle \mathbf{x}_w, \sum_{v\in V}\mathbf{x}_v \right\rangle. \quad (2)$$

Thus, we efficiently obtain the first medoid $\hat{r}_1 = \arg\max_{w\in V} g(w;\emptyset)$ with a computational cost $O(NS)$. In our approach, we employ arbitrary similarity definitions without restricting the cosine similarity. One computational advantage of using the cosine similarity is that we can efficiently obtain $\hat{r}_1$, as described above.

4.2 Acceleration of Clustering

As mentioned above, when number N of the functional vectors is large and the memory space is inadequate, we need to recalculate most of the $N(N-1)/2$ distances for each of the K (> 2) greedy steps at the k-medoids clustering phase, which amounts to a computational cost of $O(KN^2S)$. To overcome this problem, we propose a new technique for accelerating the k-medoids clustering phase by combining the lazy evaluation and pivot pruning techniques.

In the lazy evaluation technique [13], which is applied at the k-th medoid selection step, we utilize an upper bound value $UB(w)$ of marginal gain $g(w;R)$ for each candidate node $w \in V$. After initializing $UB(w) \leftarrow g(w;\emptyset)$, which is calculated in Eq. (2), we update $UB(w) \leftarrow g(w;R_h)$ when $g(w;R_h)$ is actually calculated at the h-th medoid selection step. Evidently, due to the submodular property, it is guaranteed that $g(w;R_k) \le UB(w)$ for $k > h$. Let g_k^* be the current best marginal gain at the selection step for obtaining the k-th medoid, we can then omit the calculation of $g(w;R_k)$ when $UB(w) \le g_k^*$. On the other hand, to obtain a better g_k^* at an early stage, we evaluate these candidates from the top of the sorted list by sorting the candidates nodes in descending order with respect to $UB(w)$.

In the pivot pruning technique [28], which is applied at the actual calculation of $g(w;R_k)$, we utilize lower bound distance $LB(w,v;P)$ of distance $d(w,v)$ for examining pruning condition $\rho(w,v) \le \mu(v;R)$, where $P \subset V$ is a set of pivots described below and $d(w,v)$ is a standard Euclidean distance obtained as $d(w,v) = \sqrt{1-\rho(w,v)}$. Note that from Eq. (1), we do not add any value when pruning condition $\rho(w,v) \le \mu(v;R)$ holds. From triangle inequality, we utilize the following lower bound distance $LB(w,v;P)$:

$$LB(w,v;P) = \max_{p\in P}|d(w,p) - d(v,p)| \le d(w,v).$$

When $\sqrt{1-\mu(v;R)} \le LB(w,v;P)$, noting that

$$\sqrt{1-\mu(v;R)} \le LB(w,v;P) \le d(w,v) = \sqrt{1-\rho(w,v)},$$

pruning condition $\rho(w,v) \le \mu(v;R)$ holds without actually calculating $\rho(w,v)$. Next we introduce two types of pivots P so as that the pivot pruning technique works adequately. As the first type of pivots, we utilize the obtained medoids; after setting $P \leftarrow \{r_1\}$, we successively add the obtained medoid r_k as a pivot by $P \leftarrow P \cup \{r_k\}$.

In the second type, we select some outlier nodes in the functional vector space as pivots. More specifically, by using the first medoid, r_1, we select and add the first outlier pivot by

$$\hat{q}_1 = \arg\max_{v\in V} d(v,r_1),\ P \leftarrow P \cup \{\hat{q}_1\}.$$

Then we select and add the h-th pivot by

$$\hat{q}_h = \arg\max_{v\in V}\min_{p\in P} d(v,p),\ P \leftarrow P \cup \{\hat{q}_h\}.$$

Hereafter, we denote the maximum number of outlier pivots by H, and in our proposed algorithm, we calculate these pivots before selecting the second medoid, r_2.

Hereafter, the lazy evaluation technique, the pruning technique by medoids, and the pruning technique by the outlier pivots are referred to as Lazy Evaluation (LE), Medoids Pruning (MP), and Outlier pivots Pruning (OP), respectively. In our proposed method, we apply the LE technique prior to the pivot pruning techniques. This is because when the marginal gain calculation of $g(w;R)$ is skipped by the LE technique, we can simultaneously prune all the similarity calculations of $\rho(w,v)$ for any $v \in V$. On the other hand, in our implementation, we apply the MP technique prior to the OP technique. This is because as shown later in our experiments, at the k-medoid selection step, the combination of the LE and MP techniques achieved a reasonably high performance when k becomes large. We summarize the entire flow of our proposed algorithm as follows:

(1) Select the first medoid, r_1;
(2) Select outlier pivots $P = \{p_1, \ldots, p_H\}$;
(3) Repeat the following steps from $k = 2$ to K with k increments:
  (a) Examine the pruning conditions, LE, MP, and OP, in this order;
  (b) Calculate the similarities and marginal gains for the unpruned nodes, and extract the k-th medoid, r_k.

4.3 Characterizing and Visualizing Functional Clusters

To characterize the extracted functional clusters, we introduce the Z-score measure that simultaneously compares these functional clusters of several networks. Let $V_j = \{u \in V\ ;\ |\Gamma(u)| = j\}$ and $V_j^{(k)} = \{u \in V^{(k)}\ ;\ |\Gamma(u)| = j\}$ be the sets of nodes with degree j and those belonging to functional cluster k, respectively. By defining the degree and cluster distributions by $p_j = |V_j|/|V|$ and $p^{(k)} = |V^{(k)}|/|V|$, respectively, we can calculate the following Z-score, $Z_j^{(k)}$, with respect to degree j and cluster k:

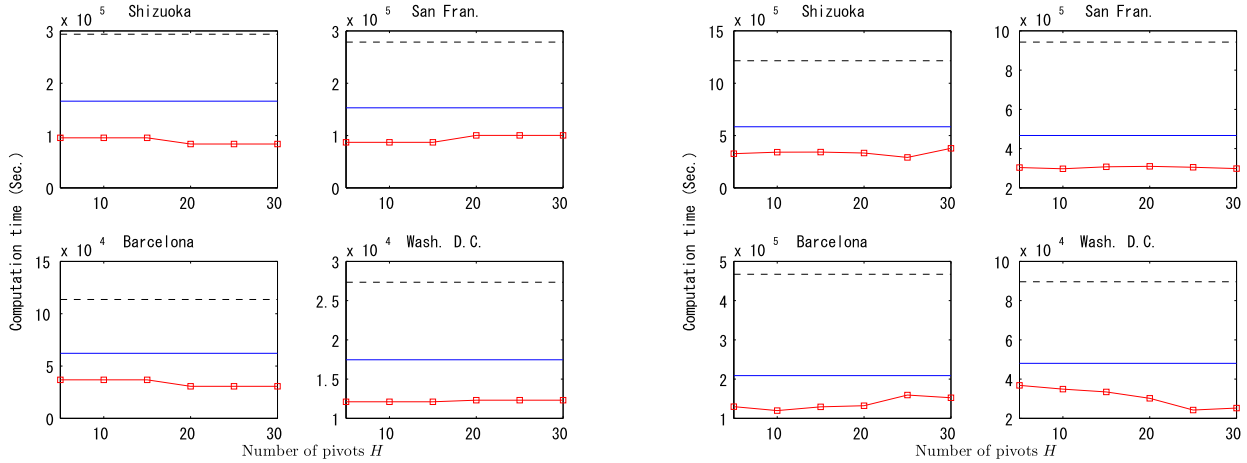
$$Z_j^{(k)} = \frac{|V_j^{(k)}| - |V|^2 \times p^{(k)} \times p_j}{\sqrt{|V|^2 \times p^{(k)} \times p_j \times (1 - p^{(k)} \times p_j)}}.$$

When $Z_j^{(k)}$ is large (or small), a significantly large (or small) number of nodes with degree j will probably appear in cluster k. In this paper, we use these Z-scores as a useful measure to characterize each functional cluster. Recall that the maximum degree of nodes in spatial networks is restricted to relatively small numbers.

In our visualization method which is based on the actual location of each node mapped from street intersections, we plot these

Table 2 Basic statistics as network.

City	Area	$\|V\|$	$\|E\|$	p_1	p_2	p_3	p_4	$p_{>4}$	C	L
Shizuoka	155 km × 119 km	110,925	162,322	.121	.070	.576	.228	.005	0.05	83.09
San Francisco	85 km × 55 km	110,700	156,821	.173	.037	.583	.198	.009	0.04	79.43
Barcelona	45 km × 30 km	66,790	99,387	.103	.031	.659	.201	.006	0.06	53.07
Washington D.C.	23 km × 18 km	24,564	38,053	.096	.028	.571	.293	.012	0.05	51.89



(a) $K = 5$ (b) $K = 10$

Fig. 2 Computation times for four cities, where black dashed lines, blue solid lines, and red solid lines with squares stand for Lazy-evaluation, Medoid-pruning, and Proposed methods, respectively.

nodes by assigning an individual color to each cluster. In our experiments we set the number of clusters to $K = 5$ and assigned green, blue, red, yellow, and magenta to $V^{(1)}, \cdots, V^{(K)}$ in this order. Here, we consistently use the same color scheme for different networks to contrast the differences of each network.

5. Experimental Data

We used the OpenStreetMap (OSM) data of the following four cities/urban areas: Shizuoka Prefecture, Barcelona and its surrounding area, San Francisco and the bay area, and Washington D.C., from Metro Extracts [*1] in August, 2015. Three of the four cities were selected as a subset of those previously studied [3], but in our experiments, each area of these cities is more than 100 times larger than the one-square mile area used in that previous study. Then we extracted all the points and lines tagged as highways from the OSM data of each city and constructed individual spatial networks by mapping the ends and the intersections of the streets into nodes and streets between nodes into links. To simplify our analyses, we deleted nodes that represent the curved segments of highways by directly connecting both sides of the deleted ones.

Table 2 shows the basic statistics of the networks for the four cities, where C and L respectively denote the averages of the clustering coefficient and the shortest path length over each network. We can see that the area and the numbers of nodes and links, $|V|$ and $|E|$, are substantially different, and that the degree distributions (defined by p_j) as well as C and L are quite similar as common characteristics of these spatial networks.

6. Experimental Evaluations

In this section, we evaluate the computational performance of our proposed algorithm and the characteristics of the extracted functional clusters.

6.1 Evaluation of the Accelerating Algorithm

We evaluate the computational efficiency of the proposed techniques for accelerating the k-medoids clustering phase under the setting of the dimensionality of functional vector $S = 10{,}000$ and number of clusters $K = 5, 10$. We compare the following three methods. The first method, which only employs the LE technique as a baseline, is called the (a) Lazy-evaluation method; the second method, which employs both the LE and MP techniques, is called the (b) Medoid-pruning method; and our proposed method, which employs all of the LE, MP, and OP techniques, is the (c) Proposed method, for which we changed the number of outlier pivots H from 5 to 30. Here we performed our experiments using a computer system equipped with an Xeon processor E5-2470 2.3 GHz and 192-GB main memory.

In **Fig. 2**, we compare the computation times of the three methods with respect to the networks of the four cities, where the horizontal and vertical axes respectively stand for the outlier pivots and the computation times. In this figure, we only show the computation times of the k-medoids clustering phase. From Figs. 2 (a) and (b) for all the networks, we confirmed that our Proposed method worked substantially faster than the Lazy-evaluation and Medoid-pruning methods. In our experimental results, we emphasize that our Proposed method achieved from three to five times better performance than the Lazy-evaluation method, which is regarded as one of the most state-of-the-art methods. On the other hand, for the desirable number of outlier pivots in our Proposed method, our experimental results indicate that it depends on the dataset. In the range of $5 \leq H \leq 30$, as shown in Fig. 2, the obtained results were almost the same. This result suggests

[*1] https://mapzen.com/data/metro-extracts

that a relatively small number of outlier pivots is reasonable in our Proposed method.

Next we evaluate the effects of the three pruning techniques in our proposed method that accelerates the clustering phase. For each k-th medoid selection step, let $LE(k)$, $MP(k)$, and $OP(k)$ be the sets of node pairs whose actual similarity calculations are skipped by the LE, MP, and OP techniques. Recall that in our proposed method, the LE, MP, and OP techniques are applied in this order. In this experiment, we set the number of pivots to $H = 30$. Thus, the actual pruning rates of the LE, MP, and OP techniques are calculated as $\alpha(LE(k)) = |LE(k)|/N^2$, $\alpha(MP(k)) = (|LE(k) \cup MP(k)| - |LE(k)|)/N^2$, and $\alpha(OP(k)) = (LE(k) \cup MP(k) \cup OP(k)| - |LE(k) \cup R(k)|)/N^2$, respectively.

In **Fig. 3**, we compare the pruning rates of the k-th medoid selection step by changing $k = 2$ to 10, where the gray, blue, and red bars respectively stand for the pruning rates of $\alpha(LE(k))$, $\alpha(MP(k))$, and $\alpha(OP(k))$. Recall that our method calculates the first medoid, r_1, by Eq. (2). From Fig. 3, for all the networks, the LE technique did not skip any marginal gain calculation at the step of $k = 2$, and it also worked quite poorly at the steps of $k = 4$. This result indicates that each upper bound $UB(w)$ was a quite rough approximation to the actual marginal gain $g(w; R)$ at these steps. The MP technique also showed relatively poor pruning rates at the step of $k = 2$. This is because just one pivot was used by the MP technique. Therefore, by applying the OP technique, our proposed method stably achieved reasonably high pruning rates.

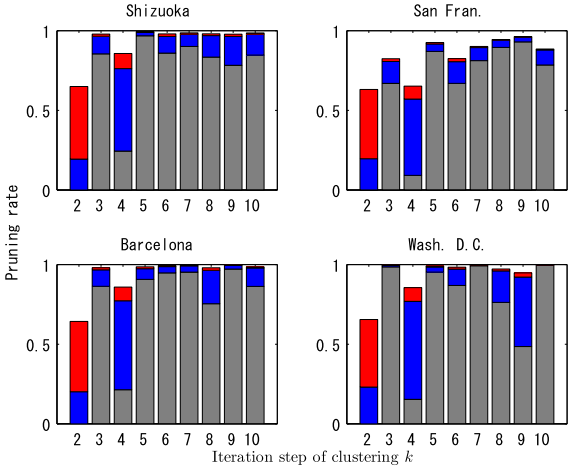
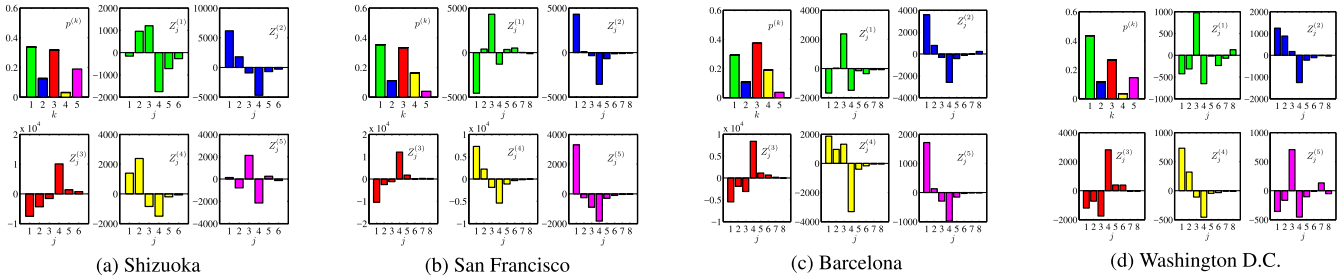


Fig. 3 Pruning rate for four cities, where gray, blue, and red bars stand for LE, MP, and OP techniques, respectively.



(a) Shizuoka (b) San Francisco (c) Barcelona (d) Washington D.C.

Fig. 4 Cluster and Z-score distributions for four cities.

6.2 Quantitative Characterization for Extracted Clusters

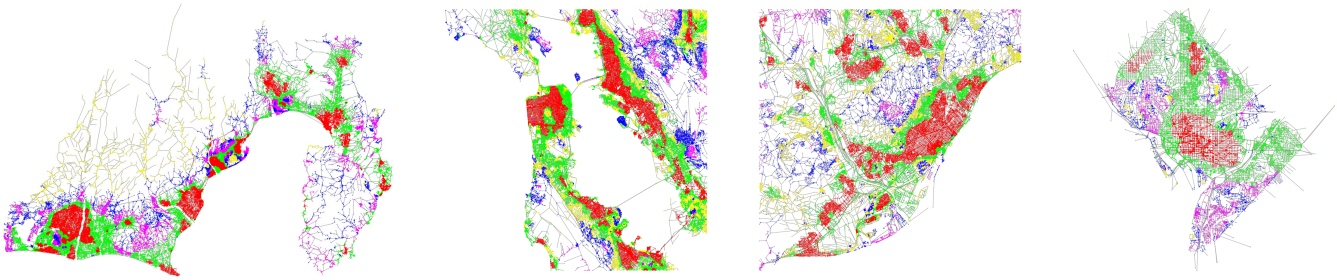
We evaluated the extracted functional clusters by our method. To this end, we characterized each of the extracted clusters by Z-scores and visualized them by actual coordinates and consistently ordered colors: green, blue, red, yellow, and magenta. **Figure 4** shows cluster distribution $p^{(k)}$, which is located at the top left in each subfigure, with the obtained Z-score distribution $Z_j^{(k)}$ for each of the four cities by setting $K = 5$, where we assigned green, blue, red, yellow, and magenta to $V^{(1)}, \cdots, V^{(5)}$ in this order, as shown below in our visualization results.

From the cluster distribution results, the union of the 1st, 2nd, and 3rd functional clusters occupied more than around 80% of the total nodes. From the Z-score distribution results, for all four cities, the 1st, 2nd, and 3rd functional clusters ($V^{(1)}$, $V^{(2)}$, and $V^{(3)}$) have common characteristics, i.e., significantly larger Z-scores for nodes with degrees $j = 3$, $j = 1$, and $j = 4$, respectively. On the other hand, the 4th and 5th functional clusters ($V^{(4)}$ and $V^{(5)}$) lack such shared characteristics. These differences might be caused by the individual characteristics of these cities, reflected by geographical restrictions and/or historical and cultural backgrounds. These characteristics of the extracted clusters can also be naturally explained by the nature of the greedy algorithm employed in the FCE method. This algorithm generally selects the first medoid r_1, creating a cluster with some average characteristics like 3-intersection regions, and then successive nodes $r_2, r_3, \ldots$ create those with such salient characteristics as cul-de-sac and lattice regions. Thus, perhaps the three former clusters ($V^{(1)}$, $V^{(2)}$, and $V^{(3)}$) reflected the common characteristics for these cities, while the two latter clusters ($V^{(4)}$ and $V^{(5)}$) reflected the individual characteristics of each city. We explained our experimental results using $K = 5$ as the minimum number, which clearly and satisfactorily demonstrates the common and individual characteristics for all four cities.

6.3 Qualitative Characterization for Extracted Clusters

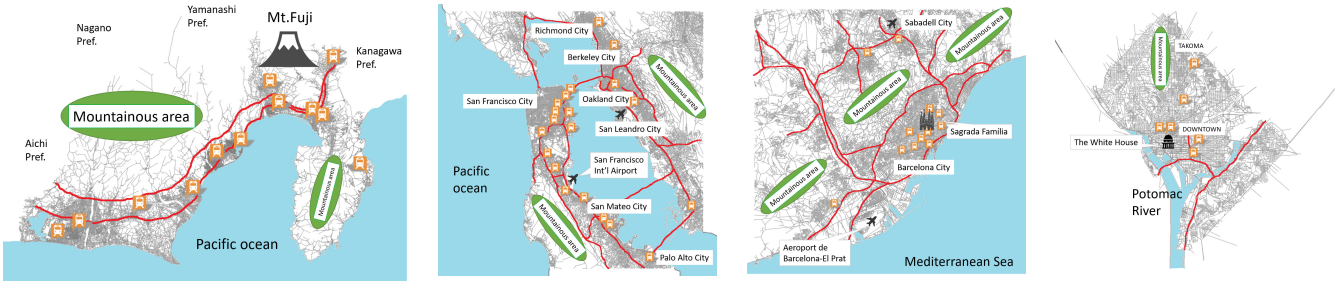
Figure 5 shows our visualization results for the four cities, where green, blue, red, yellow, and magenta are consistently used in our experiments in this paper. From these results, all four cities share the following similar characteristics: green 3-intersection regions ($V^{(1)}$), surrounded by blue cul-de-sac regions ($V^{(2)}$), and red lattice regions ($V^{(3)}$).

Figure 6 (a) indicates the main landmarks in and around the Shizuoka network, like Mt. Fuji, railway stations, mountainous areas, the Pacific ocean, branch roads, and neighboring prefectures. The red, green, magenta, blue, and yellow areas are distributed from the center of the main cities in this order. Each red



(a) Shizuoka (b) San Francisco (c) Barcelona (d) Washington D.C.

Fig. 5 Visualization results by functional clusters.



(a) Shizuoka (b) San Francisco (c) Barcelona (d) Washington D.C.

Fig. 6 Annotation with main landmarks.

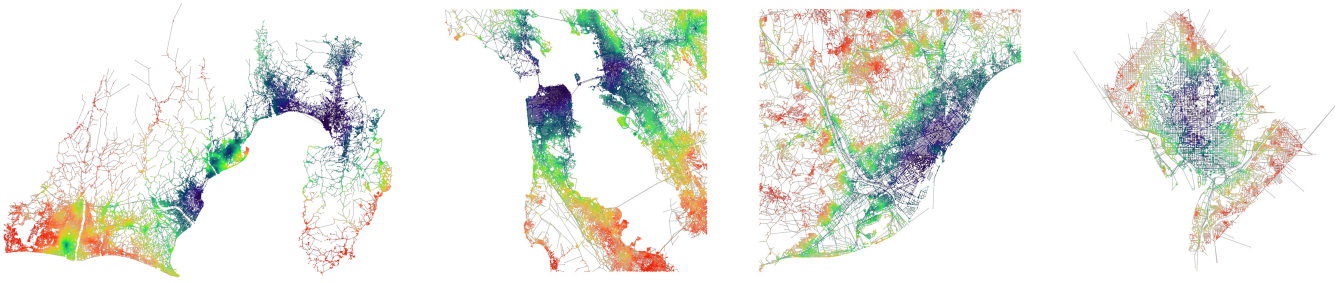
region approximately corresponds to the central area of each city, where at least one railway station exists. Also, the green regions mainly contain nodes whose degree is three and exist around each red region. Based on these observations, we call the red and green areas (functional cluster) the downtown and suburb regions. Similarly, the blue regions contain many nodes, whose degree is one, that exist in agricultural areas or at the foothills of mountains. The yellow regions contain many nodes whose degree is two, which means long continuous roads to other towns over mountainous areas. A similar tendency can be seen in other cities used in our experiments: Figs. 6 (b), 6 (c), and 6 (d). These observations, which are naturally interpretable from the aspects of geographical restrictions, suggest the practical usefulness of our method. As another advantage of our visualization results, we can intuitively understand the detailed regions of each city in terms of the characteristics of interpretable functional clusters. Therefore, our method is expected to work as a useful tool for developing and planning city promotion and travel tours as well as understanding and improving the usage of urban streets.

From Figs. 6 (b), (c), and (d), we get quite similar explanations about the results for the other three cities, as discussed for Shizuoka, especially for the first three functional clusters: $V^{(1)}$: green regions, $V^{(2)}$: blue regions, and $V^{(3)}$: red regions. Due to the property of the greedy algorithm used in our proposed method, which computes a new medoid by fixing previously selected medoids, a new functional cluster is usually formed by splitting and specifying the existing ones. For example, the 5th functional clusters ($V^{(5)}$: magenta regions) of San Francisco and Washington D.C. are respectively formed from the third one ($V^{(3)}$: red regions) and first one ($V^{(1)}$: green regions). For the 5th functional clusters ($V^{(5)}$: magenta regions) in Fig. 4, the Z-score distributions of Shizuoka and Washington D.C. are substantially different from those of San Francisco and Barcelona. Figures 4 and 5 suggest that these functional clusters of the former two cities correspond to peripheral areas for connecting city centers by 3-way junctions, while those of the latter two cities are probably mountainous areas characterized by cul-de-sacs. As mentioned earlier, the 5th functional clusters are the individual characteristics of these cities. By focusing just on the three former clusters ($V^{(1)}$, $V^{(2)}$, and $V^{(3)}$), our method can perhaps give strict definitions for such ambiguously discussed notions as the boundaries between urban and suburban areas.

6.4 Comparison to Results of Centrality Measures

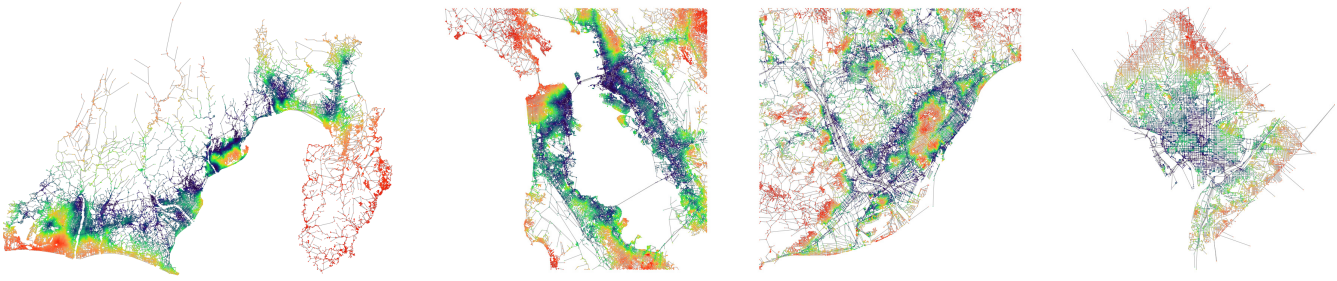
We characterize our method in comparison to three representative centrality measures: Bonacich (eigenvector), closeness, and betweenness centralities. For a given network, the Bonacich centrality ranks each node by the principal component of an adjacency matrix, the closeness centrality by the inverse of the sum of the shortest path lengths, and the betweenness centrality by the passing rate over the shortest paths between any pair of nodes [27].

Figures 7, **8**, and **9** show our experimental results using these centrality measures, where we plotted each node with a gradation color from blue to red based on the rank by each centrality measure. As common characteristics for all four cities, the Bonacich centrality typically gave high ranks to some regions (faces) of relatively high degree nodes in city centers, represented by train icons in Fig. 6. The closeness centrality gave high ranks to some streets (lines) of continuously adjacent nodes, typically on arterial roads, represented as red curves in Fig. 6, and the betweenness centrality gave high scores to some points of isolated nodes scattered widely all over the network. The highly ranked regions obtained by the Bonacich, closeness, and betweenness centrality measures were respectively represented as single types of faces, lines, and points. In contrast to these centrality re-



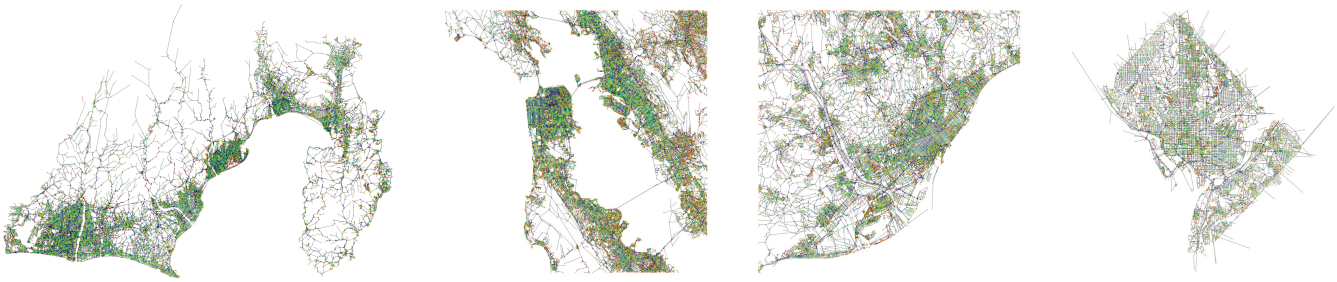
(a) Shizuoka (b) San Francisco (c) Barcelona (d) Washington D.C.

Fig. 7 Visualization results by Bonacich centrality.



(a) Shizuoka (b) San Francisco (c) Barcelona (d) Washington D.C.

Fig. 8 Visualization results by closeness centrality.



(a) Shizuoka (b) San Francisco (c) Barcelona (d) Washington D.C.

Fig. 9 Visualization results by betweenness centrality.

sults, our method extracted several types of regions represented as faces, which can be more minutely characterized in terms of interpretable functional clusters, where only the 3rd functional cluster ($V^{(3)}$: red regions) might roughly coincide with the highly ranked (blue) regions by Bonacich centrality. We empirically confirmed that our results were substantially different from those obtained by the representative centrality measures.

7. Conclusion

We addresse the problem of extracting functionally similar regions in urban streets regarding them as spatial networks. To efficiently deal with several large-scale networks, based on our previous algorithm for extracting functional clusters, we propose a new method equipped with the lazy evaluation and pivot pruning techniques for accelerating our previous algorithm with a new technique for characterizing these functional clusters and an effective way of visualizing them. In our experiments using the urban streets of four cities, we first showed that our proposed method achieves a reasonably high acceleration performance and produced a series of useful visualization results accompanied with interpretable functional clusters. We also empirically confirm that our results are substantially different from those obtained by representative centrality measures. These promising results suggest that we have also taken important steps toward tackling the interpretation problem of extracted clusters (or clustering results), which is one fundamental problem in data mining and machine learning research. In the future, we will evaluate our method using various types of networks including social networks and establish more useful tools for analyzing functional clusters.

Acknowledgments This work was supported by JSPS Grant-in-Aid for Scientific Research (No.15J00735) and (No.17H01826).

References

[1] Aggarwal, A., Deshpande, A. and Kannan, R.: Adaptive Sampling for k-Means Clustering, *Proc. 12th International Workshop and 13th International Workshop on Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pp.15–28, Berlin, Heidelberg, Springer-Verlag (2009).

[2] Burckhart, K. and Martin, O.J.: An Interpretation of the Recent Evolution of the City of Barcelona through the Traffic Maps, *Journal of Geographic Information System*, Vol.4, No.4, pp.298–311 (2012).

[3] Crucitti, P., Latora, V. and Porta, S.: Centrality Measures in Spatial Networks of Urban Streets, *Physical Review E*, Vol.73, No.3, pp.036125+ (2006).

[4] Drake, J. and Hamerly, G.: Accelerated k-means with adaptive distance bounds, *Proc. 5th NIPS Workshop on Optimization for Machine Learning*, pp.42–53 (2012).

[5] Elkan, C.: Using the Triangle Inequality to Accelerate k-Means., *Machine Learning, Proc. 12th International Conference* (*ICML 2003*), Fawcett, T. and Mishra, N. (Eds.), pp.147–153, AAAI Press (2003).

[6] Everett, M. and Borgatti, S.: Regular equivalence: General theory, *Journal of mathematical sociology*, Vol.19, No.1, pp.29–52 (1994).

[7] Fushimi, T., Saito, K., Ikeda, T. and Kazama, K.: Extracting and Characterizing Functional Communities in Spatial Networks, *Proc. Workshop on Artificial Intelligence for Tourism* (*AI4Tourism2016*), pp.182–193 (2016).

[8] Fushimi, T., Saito, K., Ikeda, T. and Kazama, K.: Functional Cluster Extraction from Large Spatial Networks, *Proc. 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining* (*ASONAM2016*), pp.57–62 (2016).

[9] Fushimi, T., Saito, K. and Kazama, K.: Extracting Communities in Networks based on Functional Properties of Nodes, *Proc. 12th Pacific Rim Knowledge Acquisition Workshop* (*PKAW2012*), Richards, D. and Kang, B.H. (Eds.), pp.328–334, Berlin, Heidelberg, Springer-Verlag (2012).

[10] Hagen, L. and Kahng, A.B.: New Spectral Methods for Ratio Cut Partitioning and Clustering, *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, Vol.11, No.9, pp.1074–1085 (online), DOI: 10.1109/43.159993 (1992).

[11] Hamerly, G.: Making k-means Even Faster, *SIAM International Conference on Data Mining*, pp.130–140 (2010).

[12] Jiang, C., Li, Y., Shao, M. and Jia, P.: Accelerating Clustering Methods through Fractal Based Analysis, *The 1st Workshop on Application of Self-similarity and Fractals in Data Mining* (*KDD2002 Workshop*) (2002).

[13] Leskovec, J., Krause, A., Guestrin, C., Faloutsos, C., VanBriesen, J. and Glance, N.: Cost-effective Outbreak Detection in Networks, *Proc. 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp.420–429, New York, NY, USA, ACM (2007).

[14] Lorrain, F. and H.White, H.: Structural equivalence of individuals in social networks, *Journal of Mathematical Sociology*, Vol.1, No.1, pp.49–80 (1971).

[15] Montis, D.A., Barthelemy, M., Chessa, A. and Vespignani, A.: The Structure of Interurban Traffic: A Weighted Network Analysis, *Environment and Planning B: Planning and Design*, Vol.34, No.5, pp.905–924 (2007).

[16] Nemhauser, G.L., Wolsey, L.A. and Fisher, M.L.: An Analysis of Approximations for Maximizing Submodular Set Functions, *Mathematical Programming*, Vol.14, pp.265–294 (1978).

[17] Newman, M.E.J.: The Structure and Function of Complex Networks, *SIAM Review*, Vol.45, pp.167–256 (2003).

[18] Newman, M.E.J.: Detecting Community Structure in Networks, *The European Physical Journal B - Condensed Matter and Complex Systems*, Vol.38, No.2, pp.321–330 (online), DOI: 10.1140/epjb/e2004-00124-y (2004).

[19] Opsahl, T., Agneessens, F. and Skvoretz, J.: Node Centrality in Weighted Networks: Generalizing Degree and Shortest Paths, *Social Networks*, Vol.32, No.3, pp.245–251 (2010).

[20] Palla, G., Derényi, I., Farkas, I. and Vicsek, T.: Uncovering the Overlapping Community Structure of Complex Networks in Nature and Society, *Nature*, Vol.435, pp.814–818 (2005).

[21] Park, K. and Yilmaz, A.: A Social Network Analysis Approach to Analyze Road Networks, *Proc. ASPRS Annual Conference 2010* (2010).

[22] Paterlini, A.A., Nascimento, M.A. and Traina, C.J.: Using Pivots to Speed-Up k-Medoids Clustering, *Journal of Information and Data Management*, Vol.2, No.2, pp.221–236 (2011).

[23] Seidman, S.B.: Network structure and minimum degree, *Social Networks*, Vol.5, No.3, pp.269–287 (1983).

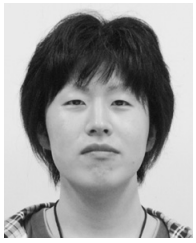
[24] Shi, J. and Malik, J.: Normalized Cuts and Image Segmentation, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.22, No.8, pp.888–905 (2000).

[25] Vinod, H.: *Integer Programming and The Theory of Grouping*, Vol.64, An Official Journal of the American Statistical Association (1969).

[26] Wang, P., Hunter, T., Bayen, A.M., Schechtner, K. and Gonzalez, M.C.: Understanding Road Usage Patterns in Urban Areas, *Scientific Reports*, Vol.2, arXiv:1212.5327 (2012).

[27] Wasserman, S. and Faust, K.: *Social Network Analysis: Methods and Applications*, Cambridge University Press (1994).

[28] Zezula, P., Amato, G., Dohnal, V. and Batko, M.: *Similarity Search: The Metric Space Approach*, Advances in Database Systems, Vol.32, 1st edition, Springer (2006).



Takayasu Fushimi was born in 1988. He received his Master of Management and Information from University of Shizuoka in 2011 and received his Ph.D. degree in Arts and Science from University of Shizuoka in 2014. He is currently an assistant professor at the School of Computer Science, Tokyo University of Technology. His current research interests are complex networks analysis and information visualization.

Kazumi Saito was born in 1963. He received his B.S. degree in mathematics from Keio University in 1985 and the Ph.D. degree in engineering from The University of Tokyo in 1998. In 1985, he joined the NTT Electrical Communication Laboratories. In 2007, he joined University of Shizuoka. He is currently a professor at the School of Management and Information. His current research interests are machine learning and statistical analysis of complex networks.

Tetsuo Ikeda received his Master of Computer Science from University of Tokyo in 1981, and the Doctor of Engineering from The University of Tokyo in 2001. In 1981, he joined the NTT Electrical Communication Laboratories. In 2002, he joined Iwate Prefectural University. In 2007, he joined University of Shizuoka. He is currently a professor at the School of Management and Information. His current research interests are data engineering, information retrieval and GIS.

Kazuhiro Kazama recieved his M.Eg. degree in precision mechanics from Kyoto University in 1988 and received his Ph.D. degree in informatics from Kyoto University in 2005. He joined Nippon Telegraph and Telephone Corporation in 1988, and joined Wakayama University in 2012. He is currently a professor at the faculty of systems engineering. His research interests are information retrieval and Web mining.