

# Detection and Classification of Pole-like Objects from Mobile Laser Scanning Data of Urban Environments

Hiroki Yokoyama<sup>1</sup>, Hiroaki Date<sup>1</sup>, Satoshi Kanai<sup>1</sup> and Hiroshi Takeda<sup>2</sup>

<sup>1</sup>Graduate School of Information Science and Technology, Hokkaido University, Japan

<sup>2</sup>Kokusai Kogyo Co., Ltd, Japan

**Abstract** – The Mobile Laser Scanning (MLS) system can acquire point clouds of urban environments including roads, buildings, trees, lamp posts etc. and enables effective mapping of them. With the spread of the MLS system, the demands for the management of roads and facilities using MLS point clouds have increased. Especially, pole-like objects (PLOs) such as lamp posts, utility poles, street signs etc. are strongly expected to be managed efficiently. We propose a method for detecting PLOs from MLS point clouds and classifying them into three classes: utility poles, lamp posts, and street signs. Our detection method is based on the feature extraction using point classification by Principal Component Analysis (PCA). On the other hand, our classification method is based on not only shape features of the PLOs, but also context features which are derived from the surrounding PLOs distributions. In order to evaluate the accuracy of PLOs detection and classification through our method, we applied our method to MLS point clouds of urban environments.

**Keywords:** Mobile laser scanning, Point cloud, Pole-like object, Object recognition, Object detection, Object classification

## 1. Introduction

With the development of inexpensive and high accuracy laser scanner devices, the Mobile Laser Scanning (MLS) system which is a vehicle mounted with these devices, GPS, INS, cameras, etc. has been widely used. The MLS point clouds can be useful not only for building and city modeling, but also for managing various facilities in urban environments. In particular, pole-like objects (PLOs) such as utility poles, lamp posts, street signs, and etc. are strongly expected to be managed efficiently, and it is required to recognize them from urban MLS point clouds. However, manually recognizing these PLOs from large point clouds data requires a great deal of time and cost. Therefore, for efficient management of facilities, it is necessary to automatically recognize PLOs from MLS point clouds of urban environment.

Much research on PLOs detection and classification from MLS point clouds has been conducted. Existing methods are based on machine learning [3], [6], [13], the arrangement and position of measurement points [7], [8], or the knowledge according to the PLO [1], [10]. These methods have some problems, such as they require a lot of training data set, they cannot recognize PLOs with different radii and tilt angles, and they are difficult to classify only using the shape features.

In this paper, we propose an algorithm to automatically detect PLOs from MLS point clouds and to classify them into three classes: utility poles, lamp posts, and street signs (Figure 1). Our detection method is based on Laplacian smoothing using the k-nearest neighbors graph, Principal Component Analysis (PCA) for recognizing points on the PLO, and thresholding for the degree of the PLO (A-1). By using smoothing and PCA, robust detection of the PLOs with various radii and tilt angles is realized. Our classification method is based on not only shape features of the PLOs, but also context features (A-2). For the shape features, we focused on the height, the number of the attached parts, and the part types of the PLO. However, in some cases it is difficult to classify PLOs only using the shape

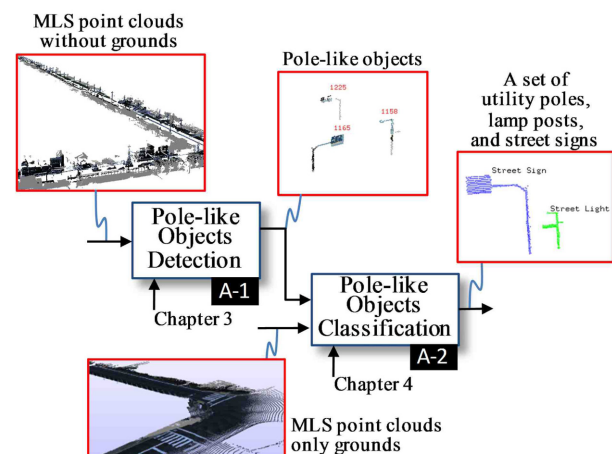


Fig. 1. Proposed PLOs detection and classification method.

\*Corresponding author:  
Tel: +81-11-706-6447  
Fax: +81-11-706-7120  
E-mail: hdate@ssi.ist.hokudai.ac.jp

features due to the similarity of the shape of the specific PLOs. For the context features, we focused on the relative position of a PLO to its local PLOs distribution. By using the context features, robust classification is realized.

Details of the existing method for PLOs detection and classification are described in chapter 2. Details of the proposed PLO detection and classification method are described in chapter 3 and chapter 4 respectively, and the results of our method are described in chapter 5.

## 2. Related Works

Existing research on automatically detecting or classifying objects, including PLOs, are introduced in this chapter.

Based on machine learning, Golovinskiy et al. proposed a method to classify various objects such as cars, streetlights, trees, fire hydrant, and etc. from the combination of MLS and ALS point clouds [3]. A major characteristic of their method is high precision segmentation by the graph cut algorithm. To classify objects, they input feature quantities of objects into the Support Vector Machine (SVM). As training data set, they used a part of the input data which are classified manually. In order to increase the recognition rate, a sufficient amount of training data set must be required. They also indicated that the better shape descriptors and classifiers are required for better results. As a similar method using machine learning, Lai et al. attempted to classify objects in MLS point clouds by using a lot of 3D data that exists on World Wide Web as training data set [6]. Ishikawa et al. proposed a method to classify objects into walls, guardrails, and PLOs using the SVM [13]. In their method, given point clouds are subdivided into the blocks, and some geometrical and statistical features are calculated in each block. Classification is done by the SVM using the calculated 31-dimensional features. In the learning based method, sufficient training data suitable for input point clouds is necessary for accurate recognition and classification.

Based on the arrangement and position of measurement points, Manandhar et al. detected vertical poles from MLS point clouds [8]. In their research, the MLS point clouds consisted of vertical scan lines and vertical poles by extracting vertical line segments from individual scan lines. Their detection method was limited to the extraction of the vertical poles. Due to this limit, the method cannot detect tilted poles and is not applicable to arbitrary point clouds. Lehtomaki et al. extracted sweeps that were expected to be measured PLOs from MLS point clouds [7]. Then they found another sweep either below the current sweep or above, and made them a cluster. However the method cannot detect PLOs with a specific radii and limited point densities.

Based on the knowledge, Lam et al. detected PLOs from MLS point clouds, and classified them into lamp

posts and utility poles [1]. Their detection method was devised based on the condition that PLOs are perpendicular to the road plane. Their classification method relied on the branch of the lamp post. However, they mentioned that some utility poles also have structure that extends from the vertical column, where the power line is attached. In such condition, it was difficult to classify the PLOs. Pu et al. detected PLOs from MLS point clouds, and classified them into bare poles, trees, traffic signs, and other poles [10]. In their classification method, the straight pole (supporter) of the PLO was removed first. When remaining point cloud was non-planar, the PLO was classified as trees. When the remaining point cloud was planar, the PLO is classified as traffic sign if the shape was rectangular or circle. If no points remained, the PLO was classified as bare pole.

In this paper, we propose a robust detection method of the PLOs with arbitrary tilt angles and radii from MLS point clouds without training data set, by using smoothing and PCA. Additionally, we propose a robust classification of the PLOs by using the shape features and context features.

## 3. Pole-like Objects Detection Method

### 3.1 Overview of the Method

The proposed algorithm is shown in Figure 2. In the method, we assume that ground points are already removed from given point clouds. The algorithm of our method consists of four steps. First, the input point clouds are segmented, as the result the points estimated on each object are grouped (B-1). Second, smoothing is applied to each segment (B-2). Third, each point is classified into the points on the PLOs, on the planar objects, and on other objects (B-3). Finally, the degree of the PLOs of each segment is evaluated, and the segments of the PLO are extracted by thresholding (B-4).

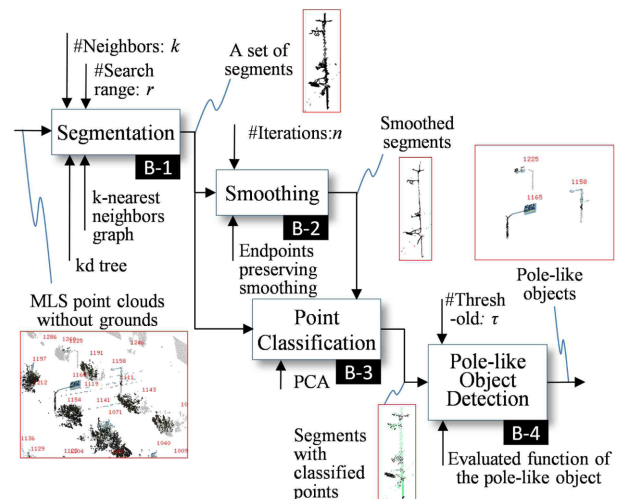


Fig. 2. PLOs detection method.

### 3.2 Segmentation

Many segmentation techniques of point clouds have already existed ([2], [9], [11]). In our implementation, we adopt the simple segmentation method by connecting the nearest neighbor points. As a result, the segment is composed of the point set of the connected  $k$ -nearest neighbors graph. The graph is generated to sequentially generate an edge among the  $k$ -nearest neighbors between the point  $i$  and points included in a sphere with a radius  $r$  centered at point  $i$ . Even if the neighbors included in the sphere are less than  $k$ , we do not expand the radius  $r$ . In our experiment,  $k = 15$ ,  $r = 0.5$  m.

### 3.3 Endpoint Preserving Laplacian Smoothing

The robust extraction of points on PLOs by PCA described in next section is difficult because of the measurement noises, the bias of the point distributions, and the differences of the PLO radius. Therefore, to improve the classification rate of the points on the PLOs by PCA, endpoint preserving Laplacian smoothing is applied to the  $k$ -nearest neighbors graph in our method.

Generally, smoothing is applied for the purpose of removing the noise of the measurement data. On the other hand, recently, smoothing is used for other purposes. For example, some methods of skeleton extraction from point clouds have been proposed [4]. The purpose of their research is to extract the skeletons of wire-objects, and the recognition of the objects is not focused on.

In our method, smoothing is applied to the MLS point clouds in order to improve the classification rate of the points on the PLOs by the PCA and to distinguish points on the PLOs from the ones on the planar objects, and the others. We focus on the exaggeration and the degeneration of the object shape features based on Laplacian smoothing which is an operation that moves each point to the center of the neighbors. Laplacian smoothing makes the PLOs into a thin pole shapes through shape degeneration. As the result, the point distributions of the PLOs come to be degenerated into a one dimensional distribution (Figure 3). In addition, measurement noises are removed. Therefore, applying the Laplacian smoothing to the scan data raises the classification rate of the PLO points by the PCA. Laplacian smoothing is done by applying Eq. (1):

$$\mathbf{p}'_i = \mathbf{p}_i + \lambda \Delta \mathbf{p}_i \quad (1)$$

where  $\mathbf{p}'_i$  is the position of point  $i$  after smoothing,  $\mathbf{p}_i$  is the position of point  $i$ ,  $\lambda$  is the smoothing strength ( $0 \leq \lambda \leq 1$ ),  $\Delta \mathbf{p}_i$  is the Laplacian, and it is given by the Eq. (2):

$$\Delta \mathbf{p}_i = \sum_{j \in i^*} \omega_{ij} (\mathbf{p}_j - \mathbf{p}_i) \quad (2)$$

	Original Mesh	After 10 times smoothing	After 1000 times smoothing
Pole-like Object			
Planer Object			
Other Object			

Fig. 3. Examples of degenerated objects in mesh model by smoothing.

where  $\omega_{ij}$  is the positive weight ( $\sum_{j \in i^*} \omega_{ij} = 1$ ),  $i^*$  is a set of the neighbors of point  $i$ . The point clouds of PLOs with various radii can be degenerated into a one dimensional distribution by iteratively applying Eq. (1), and then PLOs with various radii can be detected.

However, Laplacian smoothing has the problem that the branching structures of the PLOs are lost. This causes the decrease of the classification rate in the following PLO classification step. To solve this problem, we propose the endpoints preserving Laplacian smoothing, which controls displacements during smoothing according to the distribution of the neighbors of a point. At the endpoint, the neighbors are distributed in one direction. On the other hand, the neighbors are distributed in all directions at the inner point. From this observation, the displacements during smoothing are controlled so as to preserve the endpoints according to the distribution of the neighbors. We evaluate whether point  $i$  is the endpoint or not using Eq. (3):

$$g(i) = \frac{1}{|V_i| \times \pi} \sum_{(j,k) \in V_i} \text{angle}(j, i, k) \quad (3)$$

where  $i^*$  is a set of neighbors of point  $i$ ,  $V_i = \{(j, k) | j, k \in i^*, j \neq k\}$ ,  $\text{angle}(j, i, k)$  is the angle of  $j, i, k$ . If the point  $i$  is the endpoint,  $\text{angle}(j, i, k)$  is small (Figure 4 left), then the  $g(i)$  becomes small. On the other hand, if the point  $i$  is far from the endpoint, some  $\text{angle}(j, i, k)$  are large (Figure 4 right), then  $g(i)$  becomes large.

Displacements in smoothing can be controlled by using the smoothing strength  $\lambda$ . Additionally, in order to reduce influences of the distant points during smoo-

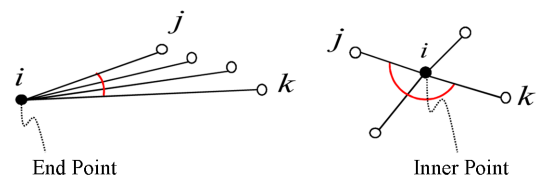


Fig. 4. Neighbors distribution at the endpoint and the inner point.

thing, we use weight  $\omega_{ij}$  which is in inverse proportion to the distance between points. Endpoints preserving Laplacian smoothing is done by the Eq. (4), (5), and (6):

$$\mathbf{p}'_i = (1-\lambda)\mathbf{p}_i + \lambda \sum_{j \in i^*} \omega_{ij} \mathbf{p}_j \quad (4)$$

$$\lambda = g(i) \quad (5)$$

$$\omega_{ij} = \frac{|\mathbf{p}_j - \mathbf{p}_i|^{-1}}{\sum_{j \in i^*} |\mathbf{p}_j - \mathbf{p}_i|^{-1}} \quad (6)$$

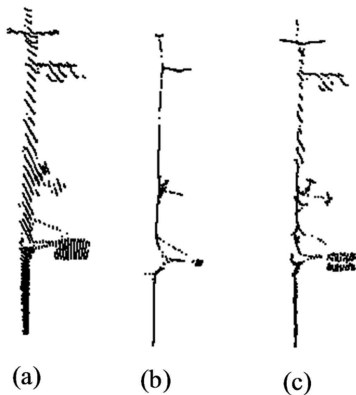
Figure 5 shows the comparative result of the Laplacian smoothing and endpoint preserving Laplacian smoothing for the same utility pole. Figure 5(a) shows the input point clouds of the utility pole. Figure 5(b) and Figure 5(c) show the results of the Laplacian smoothing and endpoint preserving Laplacian smoothing respectively. After the endpoint preserving Laplacian smoothing, the branching structures still remain. During iterating smoothing, the distribution of the points on the PLOs with various radii becomes one dimension. In our experiment, the number of iterations of the smoothing is 40.

### 3.4 Point Classification

Each point is classified into three types which are the points on the PLO, the points on the planar object, and the others. The local point distributions are evaluated by calculating eigenvalues and eigenvectors of the variance-covariance matrix related to the point  $i$  and its neighbors. The variance-covariance matrix  $\mathbf{M}_i$  of the point  $i$  is shown in Eq. (7):

$$\mathbf{M}_i = \frac{1}{|i^*|} \sum_{j \in i^*} (\mathbf{p}_j - \bar{\mathbf{p}}_i)(\mathbf{p}_j - \bar{\mathbf{p}}_i)^T \quad (7)$$

where  $\mathbf{p}_i$  is the position of point  $i$ ,  $i^*$  is a set of the neighbor points of point  $i$ ,  $\bar{\mathbf{p}}_i$  is the barycenter of  $i^*$ . We denote eigenvalues of the  $\mathbf{M}_i$  by  $\lambda_1^i$ ,  $\lambda_2^i$  and  $\lambda_3^i$  ( $\lambda_1^i \geq \lambda_2^i \geq \lambda_3^i$ ) and the corresponding unit eigenvectors by  $\mathbf{e}_1^i$ ,  $\mathbf{e}_2^i$ , and  $\mathbf{e}_3^i$  respectively. The local distribution of



**Fig. 5.** Comparative result of smoothing, (a) input point clouds of utility pole, (b) after Laplacian smoothing, (c) after endpoint preserving Laplacian smoothing (the number of iterations  $n=40$ )

neighbors of point  $i$  is figured out by the magnitude relation of the eigenvalues.

When point  $i$  is on the PLO, the maximum eigenvalue  $\lambda_1^i$  is very large compared with other eigenvalues  $\lambda_2^i$  and  $\lambda_3^i$ , and the eigenvector  $\mathbf{e}_1^i$  represents the axial direction of the PLO. On the other hand, when point  $i$  is on the planar object,  $\lambda_1^i$  and  $\lambda_2^i$  become relatively large compared with  $\lambda_3^i$ . When point  $i$  is on the other object, there are not so many differences between the three eigenvalues.

In order to investigate the magnitude relation of eigenvalues, we compute the dimensionality feature  $d_i$  using Eq. (8) [12]:

$$d_i = \arg_{j \in \{1,2,3\}} \max S_j^i \quad (8)$$

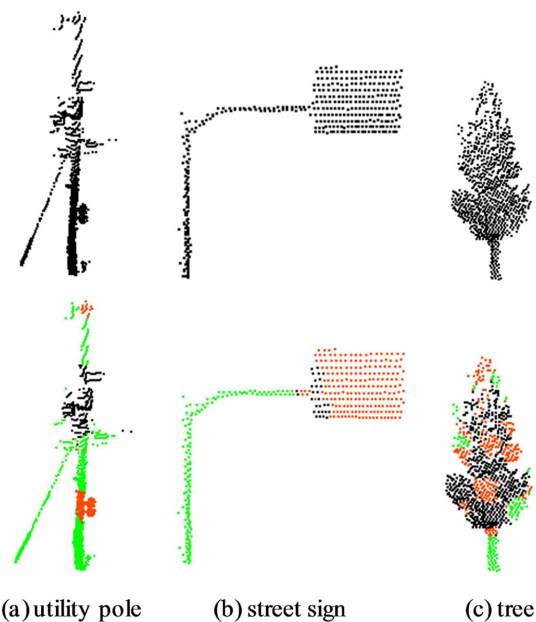
where  $S_1^i$ ,  $S_2^i$ , and  $S_3^i$  are the distribution features, and defined by Eq. (9):

$$S_1^i = \lambda_1^i - \alpha \lambda_2^i, \quad S_2^i = \lambda_2^i - \lambda_3^i, \quad S_3^i = \beta \lambda_3^i \quad (9)$$

where  $\alpha$  and  $\beta$  are the adjustment coefficient. When point  $i$  is on the PLO or planar object or other object,  $d_i$  becomes 1, 2, and 3 respectively. Figure 6 shows the results of the point classification. Figure 6(a)-(c) shows the utility pole, street sign, and tree respectively. The top row represents the input point clouds. The bottom row represents the point classification results after smoothing. In our experiment,  $\alpha$  and  $\beta$  are set to 10, 100 respectively.

### 3.5 PLOs Detection

Finally by evaluating the geometric properties and classified points of the segments, each segment is cla-



**Fig. 6.** Results of the point classification. (green: points on the PLO, orange: points on the planer object, black: points on others)

ssified into PLO and others. As the minimum requirement for the PLO, we assume that the height of the segment is more than 2 m, and that the number of points of the segment is over 50 (for removing small noise segments). Hence, the segments which do not satisfy these conditions are recognized as other type objects. In addition, the segment in which over 70% of points are the ones on the other objects is recognized as other objects, because the PLOs we intended consist of poles and plane surfaces. For each remaining segment, the degree of the PLO is evaluated by Eq. (10):

$$f_n = \left( w_1 \frac{|C_n|}{|S_n|} + w_2 \frac{|D_n|}{|C_n|} \right) \times \frac{100}{w_1 + w_2} \quad (10)$$

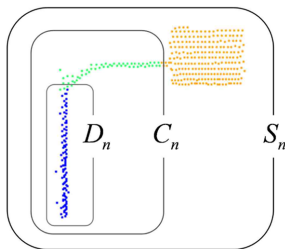
where  $w_1$ ,  $w_2$  are weights.  $S_n$  is a set of points of segment  $n$ .  $C_n$  is a set of points on the PLOs, and included in  $S_n$ .  $D_n$  is a set of points that have almost vertical  $e_1^i$ , and included in  $C_n$ . In our experiment,  $w_1 = 1.0$ ,  $w_2 = 2.0$ .

An example of the sets of points  $S_n$ ,  $C_n$ , and  $D_n$  for a street sign is shown in Figure 7. The first term of Eq. (10) represents the ratio of the points on the PLOs in the segment. The second term of Eq. (10) represents the ratio of the points whose neighbors are distributed vertically in the points on the PLOs. The degree of the PLO for each segment is evaluated by the weighted sum of the two terms. Figure 8 shows the histogram of the value  $f_n$  about various objects in MLS point clouds. The value  $f_n$  becomes larger for the almost segments of PLOs. In our experiment, because it was observed that the segments of the PLO had  $f_n$  over about 45, a threshold  $\tau$  for degree of PLO is set to 45. Finally, the segments which have  $f_n$  larger than the threshold  $\tau$  are recognized as PLOs.

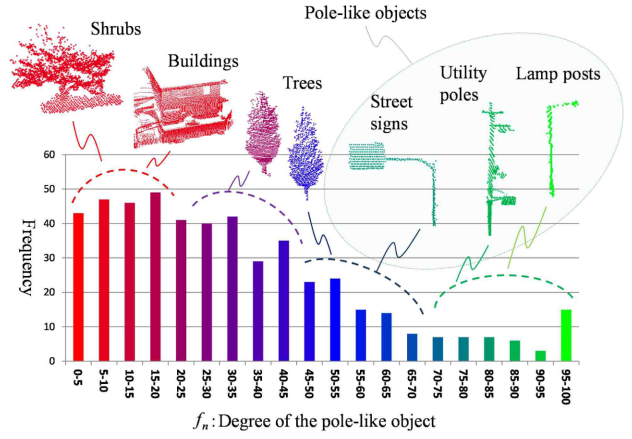
## 4. Pole-like Objects Classification Method

### 4.1 Overview of the Classification Method

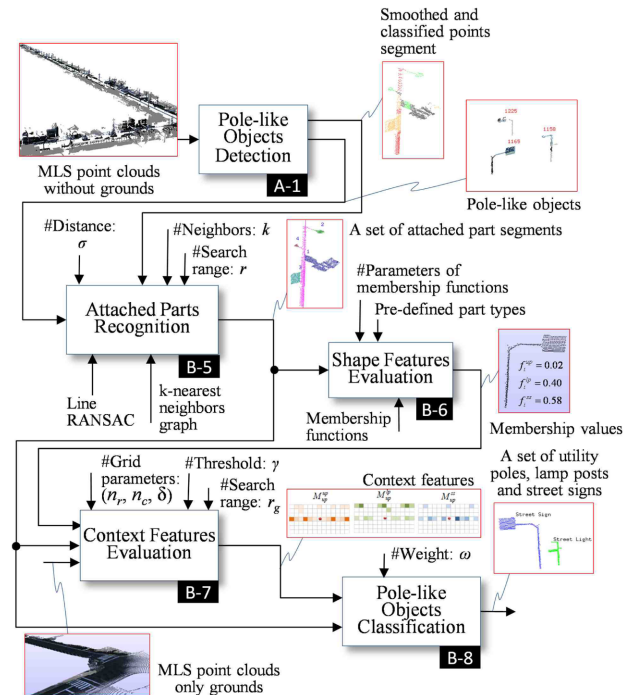
The proposed classification algorithm is shown in Figure 9. The algorithm of our method consists of four steps. First, attached parts to the supporters of the PLO are recognized (B-5). Second, the membership values based on shape features are calculated using the membership functions of the utility pole, lamp post, and street sign, which are defined by the height, the number



**Fig. 7.** Example of the point clouds of a street sign. (green: points on the PLO, orange: points on the planer object, blue: points whose neighbors are distributed vertically)



**Fig. 8.** Histogram of  $f_n$ , and objects in MLS point clouds.



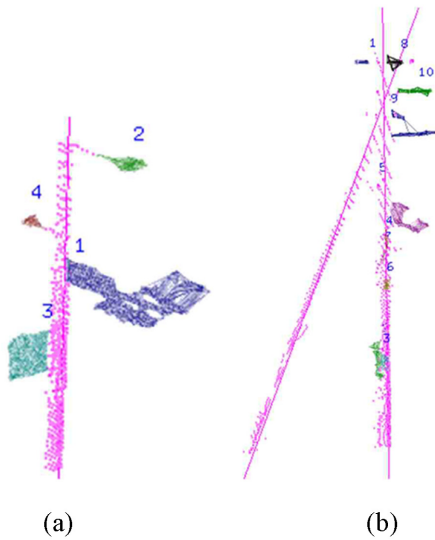
**Fig. 9.** PLOs classification method.

of the attached parts, and the attached part types of the PLO (B-6). Third, the context features using the relative position of a PLO to its local PLOs distribution are calculated (B-7). Finally, using the membership values based on shape features and context features, the PLOs are classified into utility poles, lamp posts, and street signs (B-8).

### 4.2 Attached Parts Recognition

In each PLO, the attached objects (parts) to the supporters can be the important classification cues. Therefore the supporters and attached parts are recognized from the point clouds of the PLO.

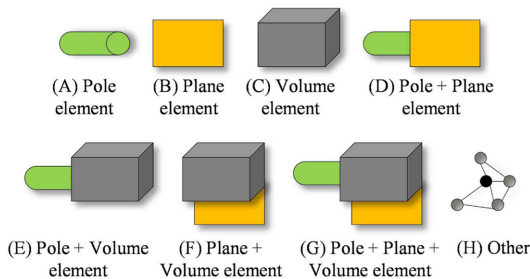
In this step, the results of the smoothing and the point classification are used (described in section 3.3 and 3.4 respectively). Line RANSAC is applied to the points



**Fig. 10.** Results of the recognition of the supporters and attached parts of the PLOs, (a) lamp post with signals, (b) utility pole composed of two supporters. (pink points: points on the supporter of the PLO, pink line: fitting line of line RANSAC, label number: attached part segment number)

whose neighbors are distributed vertically (that is, the points on the vertical pole, as shown in  $D_n$  of Figure 7). The points which lie within the distance  $\sigma$  from the fitting line are recognized as the points on the supporter of the PLO regardless of its dimensionality feature  $d_i$  (calculated in section 3.4). Figure 10 shows the results of the supporter recognition. This process is repeated until the number of points which are fitted to the line becomes less than  $\delta_1$  in order to recognize more than two supporters. Since the distances from the fitting line are calculated using points after smoothing, robust supporter recognition of the PLOs with various radii is realized. In our experiment,  $\sigma$  and  $\delta_1$  are set to  $\sigma = 0.20$  m and  $\delta_1 = 50$  respectively.

In order to analyze each attached part of the PLO, the segment for each part is extracted. First, k-nearest neighbors graph is created for the points on the PLO without the points on the supporters. Then, each connected component is recognized as a part segment. Figure 10 shows the result of the attached part segmentation. Next, the point set includes more than  $\delta_2$  ( $\delta_2 = 10$ ) and they have the same dimensionality feature  $d_i$  on the part segment are extracted. The point



**Fig. 11.** Pre-defined part types.

set is classified into pole elements, plane elements, and volume elements if  $d_i$  of points in the point set is 1, 2, and 3 respectively. Each attached part is classified into one of the pre-defined eight types from A to H shown in Figure 11 according to the kind of elements contained in the part segment.

### 4.3 Shape Features Evaluation

In our research, PLOs are classified into utility poles, lamp posts, and street signs. In many cases, PLOs have standard heights, therefore the height of PLOs become the feature for classifying. Additionally, in many cases, the utility poles have a lot of attached parts from their roles. On the other hand, the lamp posts and street signs have few attached parts. Therefore the number of the attached parts of the PLOs becomes the feature for classifying. Furthermore, the structures of the attached parts of PLOs are limited to some extent for every class, therefore the attached part types become the feature for classifying.

The membership values of the utility pole ( $up$ ), lamp post ( $lp$ ), and street sign ( $ss$ ) are tlsevaluated by Eq.(11) using shape features related to the height, the number of the attached part segments, and the part types of the PLOs:

$$f_i^{class} = f_h^{class}(h_i) + f_N^{class}(n_i) + f_T^{class}(T_i) \quad (11)$$

where  $class \in \{up, lp, ss\}$  and

$$f_h^{class}(h_i) = \begin{cases} 1.0 & h_{min} \leq h_i \leq h_{max} \\ 0.0 & otherwise \end{cases}$$

$$f_N^{class}(n_i) = \begin{cases} 1.0 & an_i + b > 1.0 \\ 0.0 & an_i + b < 0.0 \\ an_i + b & otherwise \end{cases}$$

$$f_T^{class}(T_i) = \frac{c^{class}(T_i)}{n_i}$$

Where  $f_i^{up}$ ,  $f_i^{lp}$  and  $f_i^{ss}$  are the membership values of the utility pole, lamp post, and street sign respectively. Table 1 shows the parameters of each membership function.  $h_i$ ,  $n_i$ , and  $T_i$  are the height of the PLO segment, the number of the part segments in the PLO segment and the set of the attached part types in the PLO segment respectively.  $c^{class}(T_i)$  is the number of the attached part segments in  $T_i$  which are identical to the constituent part type  $T$  of each class in Table 1. If there is no attached part,  $c^{class}(T_i)$  is 0. Parameters in Table 1 were set based on the observation of acquired MLS point clouds and catalog specification of the products. These three membership values are normalized.

However, it is difficult to accurately classify PLOs only using the membership values based on shape

**Table 1.** Parameters of the membership functions

class	$h_{min}$ [m]	$h_{max}$ [m]	$a$	$b$	$T$
up	6.5	15.0	0.1	0.0	(C, H)
lp	3.0	12.0	-0.2	1.0	(A, E)
ss	2.0	6.5	-0.3	0.9	(B, D, F, G)

features due to the similarity of the shape of the specific PLOs. To solve the problem, we introduce the context features.

#### 4.4 Context Features Evaluation

The relative position of a PLO to its local PLOs distribution is a useful cue for classification. PLOs are often found at regular intervals on a road. Therefore, typical PLOs distribution could be seen for each PLO class. From this assumption, our context features indicate the similarity between a local PLOs distribution of a PLO segment and the typical PLOs distribution of each class.

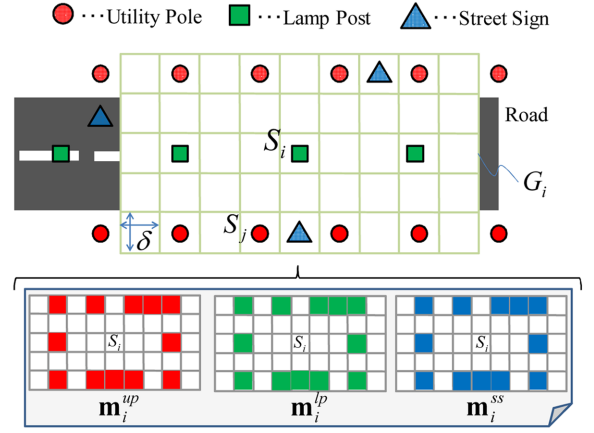
The procedure of context features evaluation consists of four steps.

i) The regular 2D-grid  $G_i$  centered at the position of the supporter of the PLO segment  $S_i$  is generated. Each cell of the grid is a square, and the length of its side is  $\delta$ [m]. The numbers of the rows and the columns are  $n_r$  and  $n_c$  respectively. Each grid direction is adjusted to the direction of the nearest road of the  $S_i$ . To find the direction of the road, PCA is applied to road point clouds within the wide range (sphere with radius  $r_g$  centered at the  $S_i$ ). The nearest road direction of the PLO is calculated as the eigenvector  $\mathbf{e}_1$  corresponding to the max eigenvalue  $\lambda_1$ . In our experiment,  $n_r = 6$ ,  $n_c = 20$ ,  $\delta = 6.0$  m, and  $r_g = 20$  m.

ii) For all PLO segment  $S_j$  on  $G_i$ , its membership values based on shape features  $f_i^{up}$ ,  $f_i^{lp}$  and  $f_i^{ss}$  are accumulated to the cell including the  $S_j$ . As a result, as shown in Figure 12, three local PLOs distribution maps (matrices)  $\mathbf{m}_i^{up}$ ,  $\mathbf{m}_i^{lp}$  and  $\mathbf{m}_i^{ss}$  which have the local PLOs membership values  $f_i^{up}$ ,  $f_i^{lp}$  and  $f_i^{ss}$  at each cells respectively are generated.

iii) Define  $U$ ,  $V$ , and  $W$  as the sets of utility poles, lamp posts, and street signs which can be clearly classified only using the membership values based on shape features. In our experiment,  $U = \{S_k | f_k^{up} > \gamma\}$ ,  $V = \{S_k | f_k^{lp} > \gamma\}$ ,  $W = \{S_k | f_k^{ss} > \gamma\}$ , where  $\gamma$  is a threshold for membership values. According to the  $U$ ,  $V$ , and  $W$ , the typical PLOs distribution maps (matrices)  $\overline{\mathbf{m}}_{true\_class}^{up}$ ,  $\overline{\mathbf{m}}_{true\_class}^{lp}$ , and  $\overline{\mathbf{m}}_{true\_class}^{ss}$  ( $true\_class \in \{up, lp, ss\}$ ) are generated as the average map of each PLO class using Eq. (12):

$$\begin{cases} \overline{\mathbf{m}}_{up}^{class} = \frac{1}{|U|} \sum_{i \in U} \mathbf{m}_i^{class} \\ \overline{\mathbf{m}}_{lp}^{class} = \frac{1}{|V|} \sum_{i \in V} \mathbf{m}_i^{class} \\ \overline{\mathbf{m}}_{ss}^{class} = \frac{1}{|W|} \sum_{i \in W} \mathbf{m}_i^{class} \end{cases}, class \in \{up, lp, ss\} \quad (12)$$



Colors on the maps represent the utility pole (red), lamp post (green) and street sign (blue) respectively, indicating that membership values based on shape features are so large that they are deep-colored.

**Fig. 12.** Local PLOs distribution maps centered at the PLO segment  $S_i$ .

The value of cell of  $\overline{\mathbf{m}}_b^a$  is the probability of existence of the object of the class 'a' when center object of the grid belongs to the class 'b'. In our experiment,  $\gamma = 0.60$ .

iv) The similarities between the local PLOs distribution map  $\mathbf{m}_i^{class}$  and the typical PLOs distribution maps  $\overline{\mathbf{m}}_{true\_class}^{class}$  are defined as the context feature, and it is evaluated by Eq. (13):

$$\begin{cases} \hat{f}_i^{up} = \sum_{class \in \{up, lp, ss\}} \langle \mathbf{m}_i^{class}, \overline{\mathbf{m}}_{up}^{class} \rangle \\ \hat{f}_i^{lp} = \sum_{class \in \{up, lp, ss\}} \langle \mathbf{m}_i^{class}, \overline{\mathbf{m}}_{lp}^{class} \rangle \\ \hat{f}_i^{ss} = \sum_{class \in \{up, lp, ss\}} \langle \mathbf{m}_i^{class}, \overline{\mathbf{m}}_{ss}^{class} \rangle \end{cases} \quad (13)$$

where  $\hat{f}_i^{up}$ ,  $\hat{f}_i^{lp}$  and  $\hat{f}_i^{ss}$  are the context features of the utility pole, lamp post, and street sign respectively. The  $\langle x, y \rangle$  is the sum of products of the values of each corresponding cell of  $x$  and  $y$ . The range of the context features are adjusted to  $[0, 1]$  by  $\hat{f}_i^{class} = \hat{f}_i^{class} / \max_{class \in \{up, lp, ss\}} \hat{f}_i^{class}$ .

#### 4.5 PLOs Classification

Finally, the PLOs are classified into utility poles, lamp posts, and street signs, using the membership values based on shape features and context features by Eq. (14):

$$S_i^{class} = \arg \max_{class \in \{up, lp, ss\}} (f_i^{class} + \omega \hat{f}_i^{class}) \quad (14)$$

where  $\omega$  represents the positive weight, and in our experiment,  $\omega = 0.25$ .

## 5. Results

### 5.1 Data and Measurement System

Point clouds of urban environment acquired by MLS system shown in Table 2 were used in our experiments

**Table 2.** Data used in the article

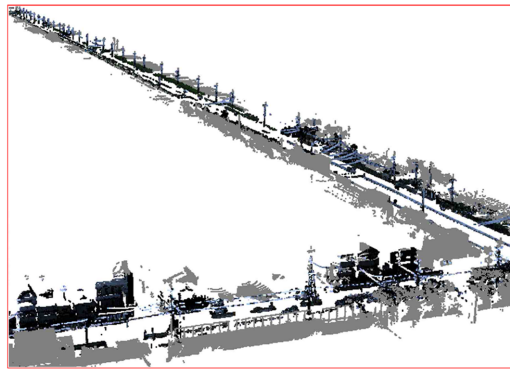
Data	Measurement Place	Number of Point (million)	Measurement System
I	Kyoto, Japan	2.5	MMS-X
II	Kyoto, Japan	1.0	MMS-X

[5]. As described in section 3.1, the ground points were removed manually from the given data. In our methods, each parameter in the algorithm is determined based on the experiments. Our method is implemented using standard PC (OS: Windows7 64bit, CPU: Intel Core i7 3.0 GHz, RAM: 6 GB).

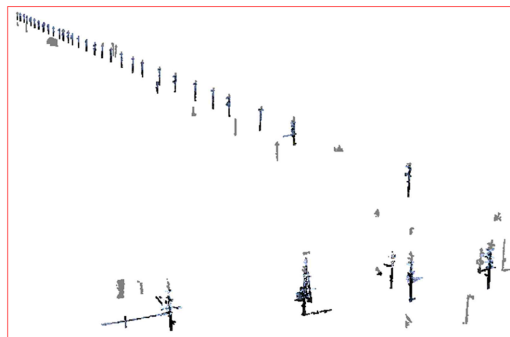
### 5.2 PLOs Detection Result

Figure 13 shows the result of the PLOs detection by our method. In Figure 13(a), there are a lot of various objects such as trees, buildings, cars, and PLOs. In Figure 13(b), the PLOs such as lamp posts and utility poles are detected. However, some other objects such as trees are included. The detection accuracy of the PLOs is shown in Table 3. The number of correct PLOs [B] was manually verified by using the photograph. [G] is the number of objects which were recognized as PLOs though they are not the PLOs. The average accuracy of the PLO detection is 69.7% for correct PLOs. On the other hand, the average accuracy of PLOs detection is 92.2% for correctly created PLOs. Some other objects such as trees are included in the false detected objects.

Our detection method is designed for correctly seg-



(a) input (Data I)



(b) output of our PLOs detection method

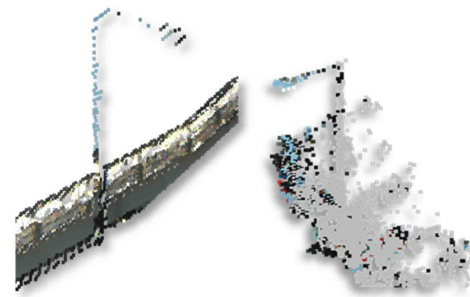
**Fig. 13.** Result of the PLOs detection.**Table 3.** Accuracy of the PLOs detection method

Data	I	II
[A] The number of total segments satisfying the minimum requirements	209	92
[B] The number of correct PLOs	63	25
[C] The number of correctly segmented PLOs	52	17
[D] The number of correctly detected PLOs	50	15
[E] Detection rate within correct PLOs ( $= [D] / [B]$ )	79.3%	60.0%
[F] Detection rate within correctly segmented PLOs ( $= [D] / [C]$ )	96.2%	88.2%
[G] The number of false detection	24	13

mented point clouds, therefore detection failed for the incorrect segments as shown in Figure 14. The [F] in Table 3 shows our method works well for the correctly created segments. In the future, we will apply an appropriate segmentation method such as [2] to the inputs and evaluate the detection rate. Processing times of the PLOs detection are shown in Table 4.

### 5.3 PLOs Classification Result

Table 5 shows the accuracy of the PLOs classification only using the membership values based on shape features. From the table, the average accuracy of PLOs classification is 66.7% for PLOs which are correctly segmented in the PLOs detection algorithm. On the other hand, Table 6 shows the accuracy of the PLOs classification using both membership values based on shape features and context features. From the table, the accuracy of classification related to the utility pole is improved. Here it should be noted that the context features could be calculated properly if there is regularity in the distribution of PLOs. In our experiment, because such regularity is seen in Data I, the classi-

**Fig. 14.** Undetectable PLOs by our detection method.**Table 4.** Running time of PLOs detection method

Data	I	II
Creation of kd tree	17.1	5.9
Segmentation	45.7	24.6
Smoothing	28.1	13.7
Point classification	17.7	17.0
PLOs detection	0.1	0.0
Total running time of detection	108.7 sec	6.12 sec



**Table 5.** Accuracy of the PLOs classification only using the membership values based on shape features\*

	Algorithm			Total	Classification Accuracy	
	<i>up</i>	<i>lp</i>	<i>ss</i>		Classification Rate	
True Class	<i>up</i>	39	15	0	54	72.2%
	<i>lp</i>	1	7	1	9	77.8%
	<i>ss</i>	0	1	1	2	50.0%
Total	40	23	2	65	66.7%	

\**up*, *lp* and *ss* represent the utility pole, lamp post and street sign respectively. The PLO segments of true class are used from [D] in Table 3.

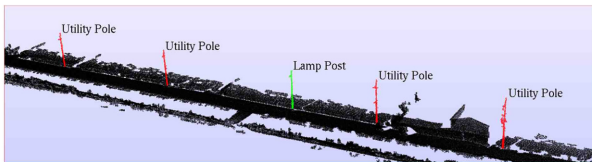
**Table 6.** Accuracy of the PLOs classification using the membership values based on shape features and context features\*

Class	Classification Accuracy	
	Only using the membership values	Using membership values and context features
<i>up</i>	39/54 (72.2%)	44/54 (81.5%)

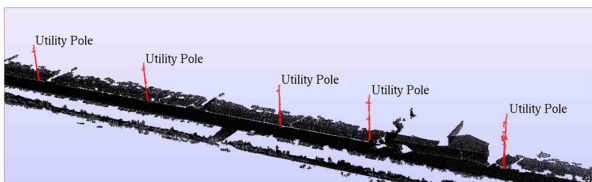
\**lp* and *ss* cannot be evaluated due to few samples.

**Table 7.** Running time of PLOs classification [s]

Process	I	II
Attached parts recognition	6.1	6.9
Shape features evaluation	23.2	6.5
Context features evaluation	1.0	0.7
PLOs classification	0.0	0.0
Total running time of classification	30.3	14.2



(a) result of the classification only using the membership values based on shape features



(b) result of the classification using both the membership values based on shape features and context features

**Fig. 15.** Effectiveness of the context features.

fication accuracy was improved. Figure 15 shows the results of the PLOs classification of Data I. Top of the figure shows the result only using the membership values based on shape features, on the other hand, the bottom of the figure shows the result using not only the membership values based on shape features but also context features. From comparison of the results, context features work well. Processing times of the PLOs classification are shown in Table 7.

## 6. Conclusion

In this article, we developed an algorithm for automatically detecting PLOs with tilt angles and various radii from MLS point clouds of urban environments, and classifying them into three classes: utility poles, lamp posts, and street signs. Our detection method is based on the smoothing and principal component analysis for point clouds, and the evaluation of the degree of PLOs for the segments using the point classification result. The PLOs detection rates were 92.2% on average for correctly segmented PLOs, and 69.7% for all PLOs including incorrectly segmented ones. Our classification method is based on shape features and context features of the PLOs. In our method, the shape feature of each PLO is calculated from the height, the number of the part segments, and part types of the PLO. Context features are calculated from the relative position of a PLO to its local PLOs distribution. By only using the membership values based on shape features, the average accuracy of PLOs classification was 66.7% for PLOs which are correctly segmented in the PLOs detection algorithm. By using both membership values and context features, the accuracy of the classification was improved to 81.5% related to the utility pole.

Future works are to improve the detection and classification rate by adopting or developing an appropriate segmentation method, and to apply our method to various MLS point clouds and evaluate its performance.

## Acknowledgement

We would like to thank Ishikawa Kiichiro of Waseda University for providing the MLS point clouds.

## References

- [1] Lam, J., Kusevic, K., Mrstik, P., Harrap, R., Greenspan, M. (2010). Urban Scene Extraction from Mobile Ground Based LiDAR Data, *5th International Symposium on 3D Data, Processing, Visualization and Transmission*.
- [2] Golovinskiy, A., Funkhouser, T. (2009). Min-Cut Based Segmentation of Point Clouds, *IEEE Workshop on Search in 3D and Vision*.
- [3] Golovinskiy, A., Kim, V., Funkhouser, T. (2009). Shape-based Recognition of 3D Point Clouds in Urban Environments, *International Conference on Computer Vision*, pp. 2154-2146.
- [4] Au, O., Tai, C., Chu, H., Cohen-Or, D., Lee, T. (2008). Skeleton Extraction by Mesh Contraction, *Proceeding of ACM SIGGRAPH*, 27, 3.
- [5] Ishikawa, K., Amano, Y., Hashizume, T., Takiguchi, J., Shimizu, S. (2009). City Space 3D Modeling using a Mobile Mapping System (in Japanese), *The Society of Instrument and Control Engineers Trans. on Industrial Application*, 8, 17, 133-139.
- [6] Lai, K., Fox, D. (2009). 3D Laser Scan Classification Using Web Data and Domain Adaptation, *Robotics*.

Science and Systems.

- [7] Lehtomaki, M., Jaakkola, A., Hyyppa, J., Kukko, A., Kaartinen, H. (2010). Detection of Vertical Pole-Like Objects in a Road Environment Using Vehicle-Based Laser Scanning Data, *Remote Sensing*, 2, 641-664.
- [8] Manandhar, D., Shibasaki, R. (2001). Feature Extraction from Range Data, *Proceeding of the 22<sup>nd</sup> Asian Conference on Remote Sensing*, 2, 1113-1118.
- [9] Moosmann, F., Pink, O., Stiller, C. (2009). Segmentation of 3D Lidar Data in Non-flat Urban Environments using a Local Convexity Criterion, *Proc. IEEE Intelligent Vehicles Symposium*, 215-220.
- [10] Pu, S., Rutzinger, M., Vosselman, G., Elberink, O., S. (2011). Recognizing Basic Structures from Mobile Laser Scanning Data for Road Inventory Studies, *Journal of Photogrammetry and Remote Sensing*, 66, 6, 28-39.
- [11] Wang, L., Chu, H. (2008). Graph Theoretic Segmentation of Airborne LiDAR Data, *Proc. SPIE Defense and Security Symposium*, 69790N-1-10.
- [12] Demantke, J., Mallet, C., David, N., Vallet, B. (2011). Dimensionality Based Scale Selection in 3D Lidar Point Clouds, *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Laser Scanning 2011*.
- [13] Ishikawa, K., Tonomura, F., Amano, Y., Hashizume, T. (2012). Recognition of Road Objects from 3D Mobile Mapping Data, *Proc. Asian Conference on Design and Digital Engineering 2012*.

---

**Hiroki Yokoyama** received his master degree of Information Science and Technology from Hokkaido University, Sapporo, Japan, in 2013. His research interest includes object recognition from point clouds and geometric modelling.

---



---

**Hiroaki Date** is an associate professor of graduate school of information science and technology at Hokkaido University. He received his doctoral degree of engineering from Hokkaido University in 2003. His research interest includes point cloud data processing, geometric modelling, and finite element mesh generation.

---



---

**Satoshi Kanai** is a full professor of systems science and informatics division in graduate school of information science and technology at Hokkaido University. He was an assistant professor in Hokkaido University from 1987-1989, and an associate professor in Tokyo Institute of Technology from 1989-1997. He is directing the projects of "Digital Ergonomics" and "Massive Digital Geometry Processing" research as a lab chair. His research interest includes Digital Prototyping, Virtual Reality and 3D Geometric Modeling. He is currently an Editor-in-chief of *International Journal of CAD/CAM*, and an executive scientific board of the *International Journal of Interactive Design and Manufacturing*. He is a member of JSPE, JSME, JSDE, IEEE and ACM.

---



---

**Hiroshi Takeda** received his Degree of Doctor of Engineering at the University of Tokyo, Japan. His is a research engineer for Kokusai Kogyo Co., LTD. His research interest includes computer vision and lidar data analysis, its data fusion, and machine learning algorithm for geo-spatial information.

---



Hiroki Yokoyama



Hiroaki Date



Satoshi Kanai



Hiroshi Takeda