S & M 2010

Identifying Staying Places with Global Positioning System Movement Data Using 3D Density-based Spatial Clustering of Applications with Noise

Nahye Cho and Youngok Kang*

Department of Social Studies, Ewha Womans University,
52 Ewhayeodae-gil, Seodaemoon-gu, Seoul 03760, South Korea

(Received April 19, 2019; accepted September 13, 2019)

Keywords: GPS log, 3D DBSCAN, movement data, staying place

In this study, we visualize and analyze global positioning system (GPS) data to identify the spatiotemporal characteristics of moving and staying patterns. As a case study, we collect and process GPS data generated by students participating in inquiry-based fieldwork. Space-time path (STP) analysis is applied to visualize movement, while density-based spatial clustering of applications with noise (DBSCAN) is used to identify spatial clusters or staying places (sites where people spend time, such as homes and workplaces). We find that some clusters derived by DBSCAN are not actual clusters, and the times spent in some clusters are overestimated when we investigate the time spent in each cluster. To resolve this, 3D DBSCAN is used to find precise clusters. The results show that the 3D DBSCAN method is effective in finding clusters of spatiotemporal data. The 3D DBSCAN methodology proposed in this study can be applied effectively in movement data analysis, such as tourist travel patterns through SNS, trajectories of cars, vessels, or wildlife, and the movement of visitors in parks.

1. Introduction

Owing to recent increases in mobile data and web usage, it has become easier to use the log data recorded during daily life; many studies have analyzed these data to find significant information. Log data created from various sensors contain spatiotemporal information; however, it is not easy to understand the characteristics of these data because the volume of data is large as well as its structure is very complex. Visualizing spatiotemporal data plays an important role in discovering data characteristics. A typical method of visualizing 3D spatiotemporal data is space-time path (STP) analysis.(1–3) Zhao *et al.* used STP to visualize the movement behavior of tourists visiting New Zealand and analyzed which activities (e.g., sightseeing, vacation, and dining) tourists were performing at staying places.(2) Zhao *et al.* visualized Halifax census tract data according to individual daily movement patterns.(1) Ostermann showed that it was possible to trace the movement paths of park visitors and identify places with a high density of activities by using global positioning system (GPS) location information.(4)

*Corresponding author: e-mail: ykang@ewha.ac.kr

https://doi.org/10.18494/SAM.2019.2410

ISSN 0914-4935 © MYU K.K.
https://myukk.org/

In addition, several studies have been conducted using GPS data to analyze users' positions and extract staying places such as user's homes and workplaces(5) or points of interest (POIs)(6,7) by considering movement and time information. Other studies have analyzed user movement paths to understand preferred routes or find similar trajectories,(8,9) proposed systems for identifying major places for each user and recommending activities that can be done at these places,(10) and proposed a methodology for predicting a user's next destination using their GPS data.(11,12) Clustering algorithms are commonly used in these studies to identify the staying places of groups.(6,10,11,13–16) The density-based spatial clustering of applications with noise (DBSCAN), which can distinguish areas with high density from those with low density within a constant spatial range, is widely applied.(17,18)

However, the DBSCAN method has a limitation for analyzing spatially dense regions because it does not consider the time dimension. In this study, we propose a 3D DBSCAN methodology to identify clusters of spatiotemporal data. To do this, we apply our 3D DBSCAN methodology with the GPS data collected from inquiry-based fieldwork and analyze how 3D DBSCAN results differ from regular DBSCAN results.

2. Materials and Methods

2.1 Log data collection and processing

In this study, we analyze log data collected during an inquiry-based fieldwork program, "sustainable development of Yangdong Village," a location registered as a World Heritage Site for its traditional Korean Hanok houses. Although many buildings have been preserved, some have been converted from residential to commercial use and from traditional to modern style. Student activities at Yangdong village include three types: making basic observations of whether an individual house has been changed or not, performing an activity requiring a judgment on whether a house should be allowed to change or must be preserved, and conducting interviews with residents. The house-based investigation takes little time, but the activities requiring students' judgment or interviews with residents take significantly longer. The students use Collector for ArcGIS on mobile devices to record their answers to inquiry questions, and the results are saved in the cloud-based ArcGIS Online in real time. The Collector for ArcGIS application does not track and store students' movement logs.(19) Therefore, a multi-camcorder equipped with a GPS device is used for this purpose. Table 1 shows the environment for collecting movement logs as well as the equipment and tools used to collect log data. A movement log contains the time, latitude, longitude, altitude, speed, and slope at which field

Table 1
Experimental environment.

Experimental environment	Place	Gyeongju Yangdong Village
	Time	17/06/2016 (about 150 min)
	Participants	17 students, 8 teams
Movement log	Tracking device	Multi-camcorder (GPS)
Answers	Observation tool	Collector for ArcGIS

activities are performed, as well as errors. This information is saved in the .nmea file format. Figure 1 shows an example of 2D GPS data.

The movement log, once collected, is processed in three steps, as shown in Fig. 2. First, the .nmea file, which is the log data collected from the multi-camcorder, is converted into a .txt file by the GPS Visualizer program, which in turn is converted into a .csv file. Second, the csv file is preprocessed by removing data beyond the spatial range or with errors. Third, the .csv file is converted into a .shp file, in which a spatial coordinate system is defined and the time fields of attribute data are converted into a date format (yyyy-mm-dd hh:mm:ss) and then a numeric format, which can be recognized by GeoDataBase (GDB) and R. Finally, data analysis is performed in three steps, namely, visualizing the log data 3D format with STP, locating staying places using DBSCAN, and checking the clusters with 3D DBSCAN considering the time dimension.

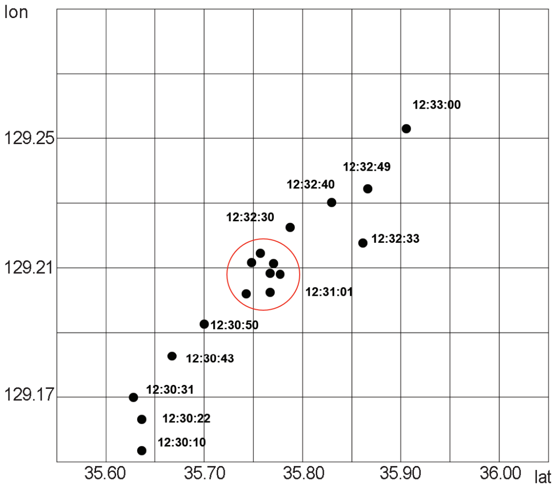


Fig. 1. (Color online) Example of GPS data: GPS data are transmitted regularly every 10 to 15 s. Here, the movement is from southwest to northeast.

Data Creation
Data Transformation
Data Processing
Data Analysis

Log data
- Spatial data (X,Y coordinate)
- Attribute data (date, time, altitude etc.)

Video data
- Field activities' type, Field activities' spot, Places where each team stayed

Log data
- .NMEA (from GPS) to .TXT
- .GPX (from app) to .TXT
- .TXT to .CSV

1) Spatial data
- .CSV to .SHP
- Define Projection

2) Attribute data
- Time data type:
  ① string to date (yyyy-mm-dd hh:mm:ss)
  ② date to numeric

Data selection
- Set the necessary items of log

Data cleansing
- Removing unnecessary data

① The XY value is out of the survey range
② Error is large due to wifi, cellular data, etc.
③ In case of indoor, there is a large error in GPS data
④ GPS errors caused by buildings

3D visualization
- Create and visualize the STP (R rgl pacage etc., ArcScene)

Finding of staying places with DBSCAN
- Analysis of dense regions in field experience learning using DBSCAN

Check clusters using 3d DBSCAN
- Analysis of considered time dimension using 3D DBSCAN

Fig. 2. Data processing steps.

2.2 Spatiotemporal visualization of movement logs using STP

To visualize the log data, the STP technique is applied. This technique is based on the concept that time can be measured in the same way as distance.(20,21) A person's movement path can be shown on a 2D plane; however, in this concept, a time axis is added, and the movement path's position at each time interval is shown in a 3D space to create a STP.(22) When convergence occurs at a particular point on the STP, that point is called a station, which is a place where activities are focused over a fixed time period, such as a school, a house, or a restaurant. The open-source R3.5.0 program is applied to create STPs because R has scatterplot3d and rgl packages that can create STPs in addition to simple 3D point mapping. Longitude, latitude, and time data are entered as the *x*, *y*, and *z* coordinates, respectively. After performing the scatterplot3d() and plot3d() functions, these factors are changed into a continuous plot in STP form. The source code used in this study is shown in Table 2.

2.3 Methodology for identifying staying places

To locate staying places, it is necessary to distinguish movement from stopping in the GPS point data. When movement–stopping patterns are visualized as STPs, parts parallel to the time axis are assigned as "stopping," and spatial clusters derived by the cluster analysis of the point data are classified as staying places. To distinguish movement from stopping and to find staying places, many previous studies have used clustering methods. Of the various forms of clustering, typical algorithms for finding staying places include the *k*-means algorithm,(11) fuzzy clustering algorithm,(10) and grid-based clustering methodologies.(10,13,14) The *k*-means and fuzzy clustering algorithms have been used many times, but they have the drawback of requiring the number of clusters to be set beforehand, following the assumption that each point must belong to a specific cluster. Grid-based clustering methodologies are suitable for finding clusters considering the spatial dimension, but they have the drawback of not being suitable for detecting POIs and movement paths in a more detailed way. In this study, we use DBSCAN,

Table 2
3D visualization source code.

R-rgl, scatterplot package

```
library(rgl)
library(scatterplot3d)

//Input spatiotemporal data
x <- longitude
y <- latitude
z <- time

//Visualize data multidimensionally
plot3d(x, y, z, xlab = "Longitude", ylab = "Latitude", zlab = "Time", main = "Fieldwork", type = "l",
ticktype = "detailed", col = c(team), expand = 1.03)
scatterplot3d(x, y, z, xlab = "Longitude", ylab = "Latitude", zlab = "Time", main = "Fieldwork", mar =
c(4,3,2,3), type = "p")
s3d$points(x, y, z, col = c(loc$team))
```

which is a density-based clustering methodology and has the advantage of having a data-specific clustering form.

The core idea behind DBSCAN is that points are assigned to the same cluster if they arrive at a fixed density. The concept of DBSCAN comprises the three definitions shown below.(23,24)

[Definition 1] ε-neighborhoods:

The ε-neighborhood $N_\varepsilon(p)$ of a data point p is the set of points within a specified radius ε around p:

$$N_\varepsilon(p) = \{q \mid d(p, q) < \varepsilon\}, \tag{1}$$

where d is some distance measured and $\varepsilon \in \mathrm{R}^+$. Note that point p is always in its own ε-neighborhood; that is, $p \in N_\varepsilon(p)$ always holds. DBSCAN uses $N_\varepsilon(p)$ and a threshold called *minPts* to detect dense regions and classify the points in a data set into core, border, or noise points.

[Definition 2] Point classes:

A point $p \in D$ (a domain) is classified as follows: (a) a core point if $N_\varepsilon(p)$ has high density; that is, $|N_\varepsilon(p)| \geq minPts$ where $minPts \in \mathbb{Z}^+$ is a user-specified density threshold; (b) a border point if p is not itself a core point, but is in the neighborhood of a core point $q \in D$, i.e., $p \in N_\varepsilon(q)$; or (c) a noise point, otherwise.

[Definition 3] Directly density-reachable points:

A point $q \in D$ is directly density-reachable from a point $p \in D$ with respect to ε and *minPts* if and only if

$$N_\varepsilon(p) \geq minPts \tag{2}$$

and

$$q \in N_\varepsilon(p). \tag{3}$$

DBSCAN's basic functions are shown in Table 3. Euclidean distance is generally used for neighbor calculations of this function. To apply DBSCAN, it is necessary to determine the adjacent radius ε and the density threshold (i.e., the minimum number of points, *minPts*). To find an appropriate ε, the points' kNN distances (i.e., the distance of each point to its kth nearest neighbor) are plotted in descending order on the basis of heuristic methods, and protruding points on the plot need to be found.

A staying place in this study is simply where students spend lots of time continuously during the fieldwork. Staying places may relate to observation or inquiry activities, or they may not. The amount of time spent at a staying place can be interpreted as showing the level of problem difficulty if the staying place is an observation or inquiry activity location. Otherwise, it can be interpreted as a place where some element hinders inquiry activity. Therefore, the time analysis of staying places is necessary.

In this study, the first GPS log included among the multiple GPS points in a staying cluster is defined as gs and the last log is defined as ge. The time difference between gs and ge is

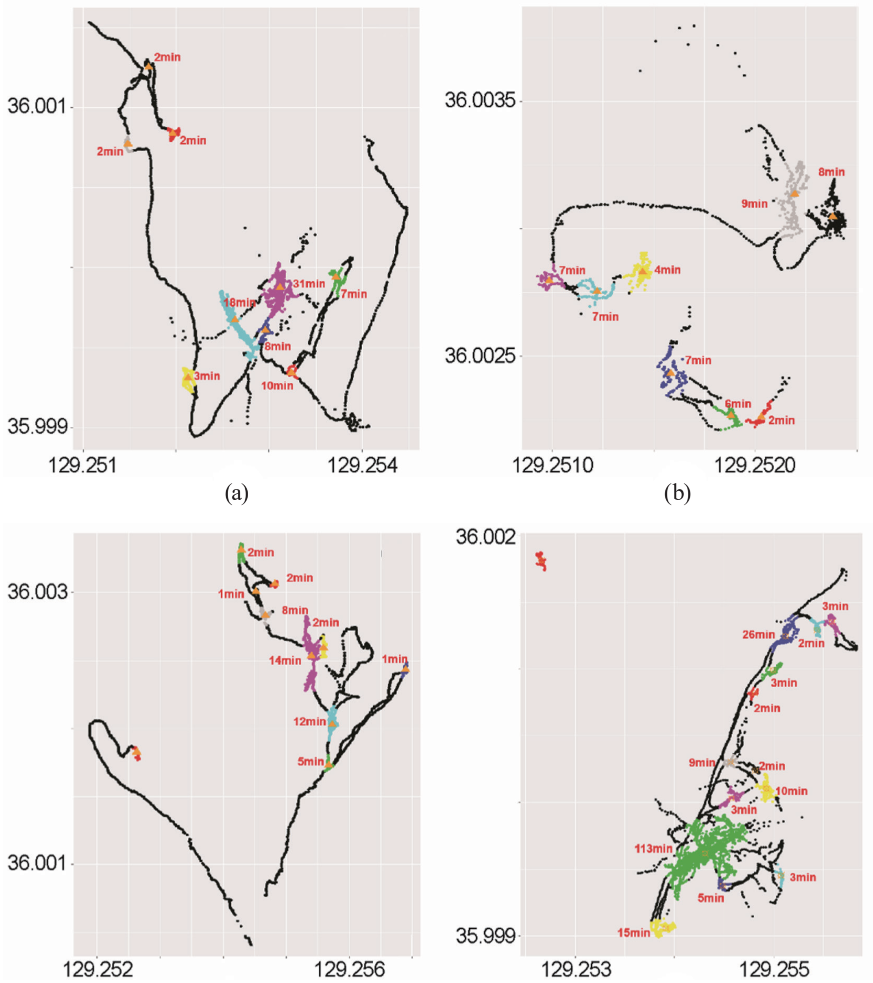
Table 3
DBSCAN analysis code.

DBSCAN analysis

```
library(dbscan)
// Input data
x <- longitude
y <- latitude

// eps, minPts
> kNNdistplot(x, k = 4)
> abline(h = .05, col = “red”, lty = 2)

// Implement dbscan
> db <- dbscan(x, eps, minPts = 4, weights = NULL, borderPoints = TRUE, ...)
> db

DBSCAN clustering for 5,684 objects.
Parameters: eps = 0.05, minPts = 4
The clustering contains 6 cluster(s) and 30 noise points.
0 1 2 3 4 5 6
30 185 87 89 3 3 3
Available fields: cluster, eps, minPts

// convex cluster
> hullplot(x, res)
```

the staying time. Time information at staying places is illustrated in Fig. 3. We find that at least 2–3 min and at most 5–8 min are spent at staying places during observation activities. However, there are also cases where more than 15 min, or 10% of the overall investigation time, is measured as the staying time. For example, when one place is visited several times with movement and staying occurring repeatedly, the temporal range of the cluster increases. As shown in Fig. 4, this place is a junction point on a path through which students moved to other places rather than a place where they stayed continuously. As the time interval for passing this point becomes larger, time becomes overestimated. In cases like this, it is necessary to apply the DBSCAN method while also considering the time dimension.

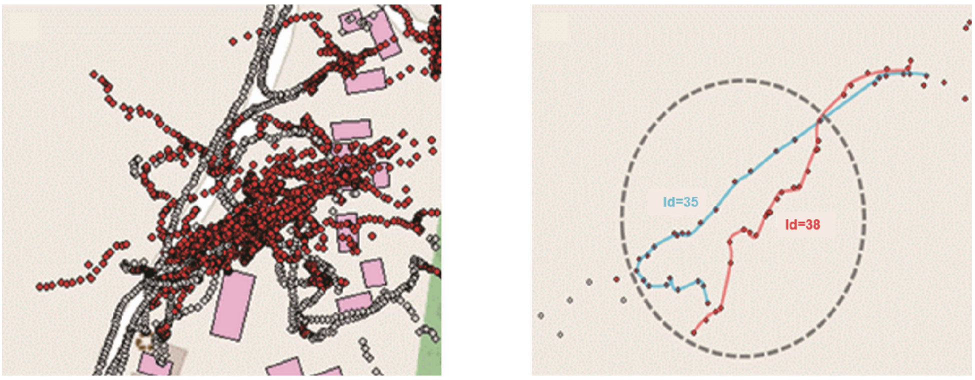
To solve the above problem, we propose a 3D DBSCAN methodology, which is a method of calculating the distance d between points p and q in three dimensions as shown in Eq. (5). Note that x, y, and z values represent latitude, longitude, and time, respectively. The x, y, and z values are incorporated in combination with the plot3d() function to identify clusters while considering time (Table 4). In this study, x, y, and z data are converted and analyzed as follows. The x and y data are converted into meters by transforming the geographic coordinate system of latitude and longitude to a projection coordinate system (EPSG: 3857 WGS_1984_Web_Mercator_Auxiliary_Sphere). The z data are converted from date type (yyyy-mm-dd hh: mm: ss) to numeric type (5 digits and decimal places). The conversion principle is that the numeric value increases by 1 as a day passes. Since 1 day is 24 h, 1 hour is about 0.0417 (1/24), 1 min is about 0.000694 [1 / (24 × 60)], and 1 s is about 0.000011574 [1 / (24 × 60 × 60)]. For example, if the z values at points p and q differ by 10 min from 2016-06-17 13:05:02 and 2016-06-17 13:15:02,



(a) (b)

(c) (d)

Fig. 3. (Color online) Example of time spent at staying places: (a) Team 2, (b) Team 4, (c) Team 6, and (d) Team 8.



(a) (b)

Fig. 4. (Color online) Example of time being overestimated: (a) the number of points is increased because the movement flow in one cluster is repeated; (b) the blue path indicates the initial movement of students in the upper right direction. The red path, after a while, indicates movement in the lower left direction.

Table 4
Source code for 3D DBSCAN.

3D-DBSCAN (DBSCAN Application)

```
library(dbscan)
// Input data
x <- longitude
y <- latitude
z <- time

// eps, minPts
> kNNdistplot(x, k = 4)
> abline(h = 0.0001, col = “red”, lty = 2)

// Implement 3D DBSCAN
> data_3d<- matrix(c(x, y, z), ncol=3)
> db<-dbscan(data_3d, eps = 0.0001, MinPts = 4)
> plot3d(x, y, z, col = db$cluster+1, pch = 20, cex = 2, main = ”Zone1, MinPts = 4”, type = “p”, ize = 5)

>db

DBSCAN clustering for 193 objects. Parameters: eps = 1e-04, minPts = 4 The clustering contains 7 cluster(s) and
143 noise points. 0 1 2 3 4 5 6 7 143 11 5 5 4 6 10 9
```

the difference is 0.006944, their numeric values being 42538.545162037 and 42538.5521064815, respectively. After conversion, the *x*, *y*, and *z* values are normalized with a 1:1:1 ratio by the min-max scaling method [Eq. (6)].

DBSCAN: distance between two points $p(x_1, y_1)$ and $q(x_2, y_2)$:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{4}$$

3D DBSCAN: distance between two points $p(x_1, y_1, z_1)$ and $q(x_2, y_2, z_2)$:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \tag{5}$$

Min-max scaling:

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{6}$$

3. Results of Analysis

3.1 Movement log visualization

Log data were created during approximately 150 min of fieldwork by eight teams with two or three students each. The STP visualization shows not only the movement distance, range, and direction, but also the staying places by creating straight lines in the *z* or time direction when the staying time at a given point is sufficient (Fig. 5). After observation or inquiry activities, students were asked to enter their answers, so most of the places they stayed match the locations where they entered their answers. That said, staying places with and without answers are distinguished. Staying places with answers can be defined as places of observation. Staying places without answers are places where other activities occur, and additional analysis is needed to identify why students stayed at those places. Figure 6 shows a generalization of the movement characteristics discovered through the visualization.

3.2 Finding staying places with DBSCAN

The most important element in using the DBSCAN method is setting the ε (minimum search radius) and *minPts* (in the area) so as to identify clusters. In this study, we extract 20% of all data at random for use as a test data set. The optimum values are found by cross-testing the ε and *minPts* values from the test data and using the kNNdistplot() function. We set arbitrary *minPts* and then apply kNNdistplot(). On average, about 50 points of log data are collected for 1 min, about 70 points per min are collected at moving intervals, and about 40 to 60 points are collected at staying intervals. Therefore, after setting *minPts* in the range of 40 to 60, we use kNN distances to find appropriate ε values. Based on cross-testing and kNNdistplot() on the test data, the critical value is 0.00005 (about a 10 m radius) for all these minimum numbers

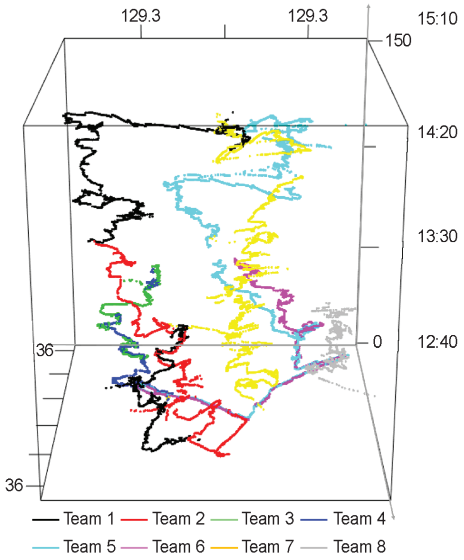


Fig. 5. (Color online) STP visualization of all movement logs. Colors denote the different teams.

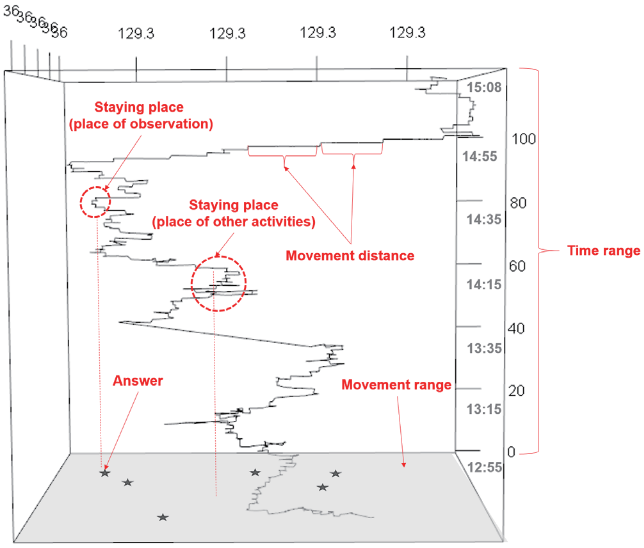


Fig. 6. (Color online) Generalized diagram of movement characteristics: If a path is vertical (i.e., parallel to the *z*-axis), it can be considered staying; other lines denote movement paths. Staying places where answers are logged can be defined as places of observation. Staying places that have no answers are defined as places where other activities occur.

of points: 40, 50, and 60 (40-NN, 50-NN, and 60-NN), as shown in Fig. 7. Table 5 shows the number of clusters based on different combinations of the two parameters.

The parameter conditions of ε = 0.00005 and *minPts* = 40 identify the largest number of clusters that are closest to actual staying places (Fig. 8). Figure 9 shows the movement of each team and clusters that represent staying places. The staying places for each team are shown in Table 6. In total, 100 clusters are identified, and the number of clusters for each team is at least 4 and at most 27. There are also 9652 noise points.

3.3 Analysis of staying places using 3D DBSCAN

GPS data were analyzed using the 3D DBSCAN method to determine whether or not the clusters generated using 2D information remain as the same cluster when considering time. Table 7 shows the numbers of clusters determined by the DBSCAN and 3D DBSCAN methods. The clusters analyzed using 3D DBSCAN are shown in two types: the numbers of clusters viewed from a 2D plan and a 3D cube. We found that the numbers of clusters extracted are smaller using the 3D DBSCAN method than using the DBSCAN method. Some clusters derived in two dimensions are no longer clusters, or else are subdivided, when the time dimension is accounted for using 3D DBSCAN.

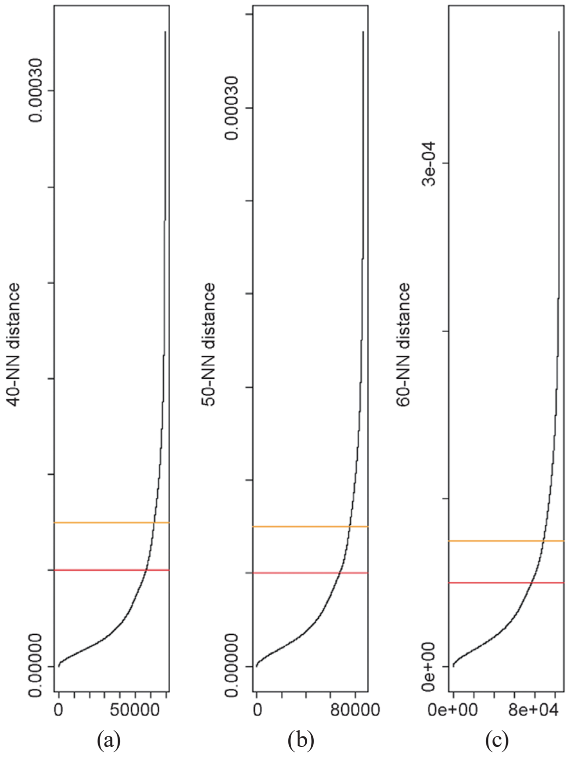


Fig. 7. (Color online) kNNdistplot() function according to different *minPts*: (a) 40, (b) 50, and (c) 60 points.

Table 5
Numbers of clusters according to different parameter combinations.

minPts	40	50	60
Critical ε	0.00005	0.00005	0.00005
Number of clusters	27	24	20

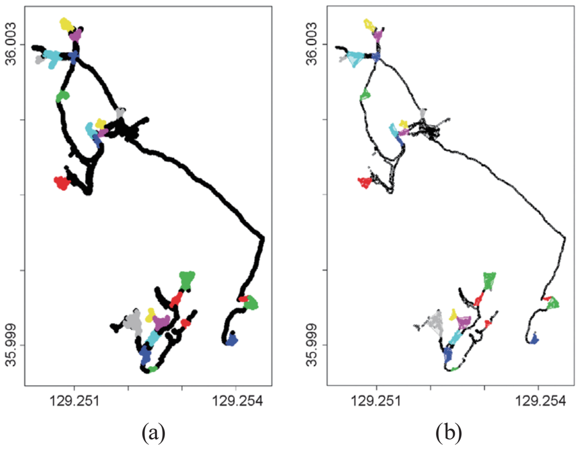
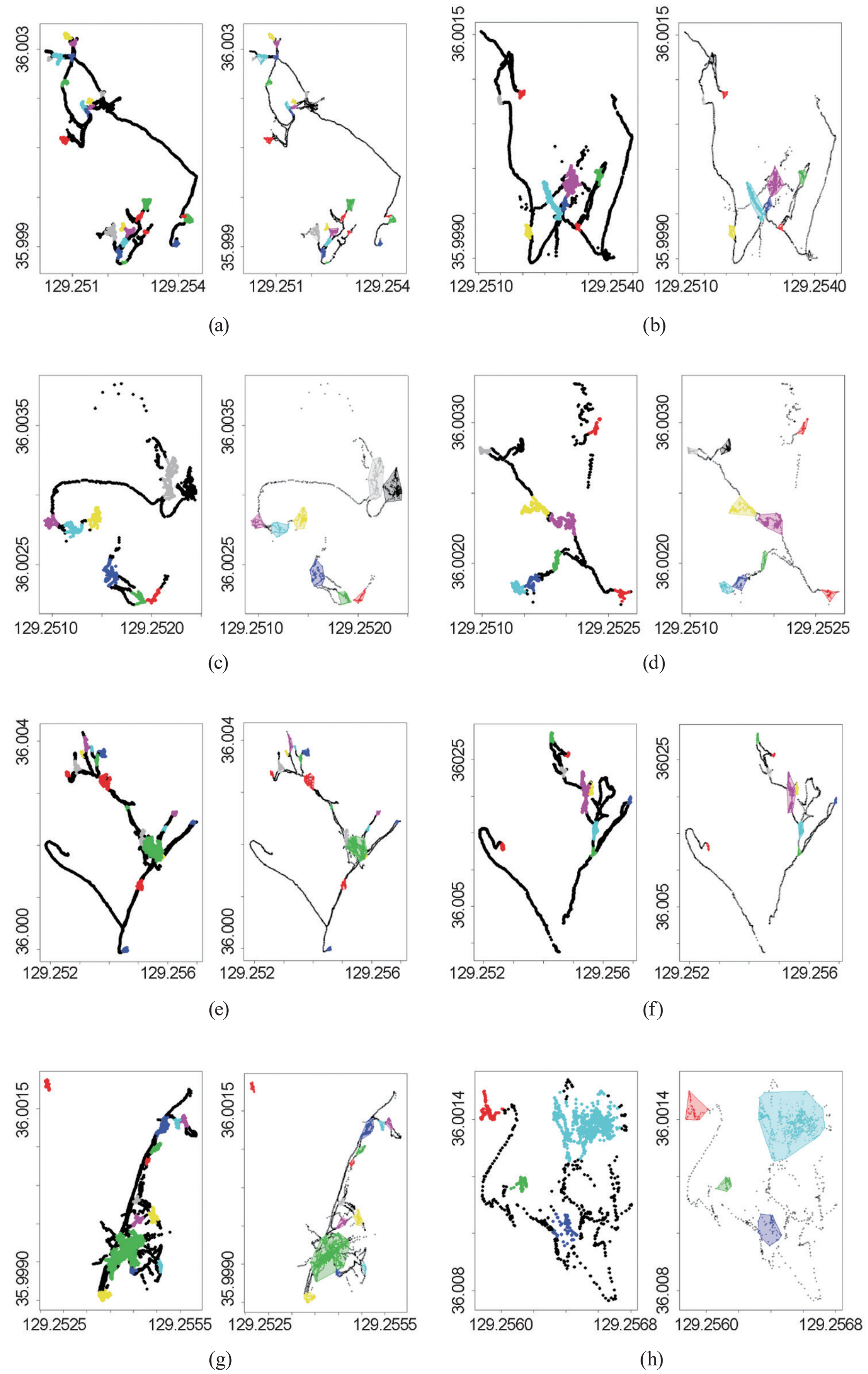


Fig. 8. (Color online) Analysis of DBSCAN results for Team 1: (a) movement is shown in black points; staying places are shown in various colors; (b) staying places are shown as convex hulls of various colors.



(a) (b)

(c) (d)

(e) (f)

(g) (h)

Fig. 9. (Color online) Visualization of movement and staying places for each team: (a) through (h) represent Teams 1 through 8, respectively.

Table 6
Number of clusters for each team using the DBSCAN method.

	T1	T2	T3	T4	T5	T6	T7	T8	Total
GPS points	6525	3549	1784	1735	6900	2948	5976	1390	30807
Number of clusters	27	9	8	9	19	10	14	4	100
Noise points	1636	1216	360	350	1617	2948	1127	398	9652

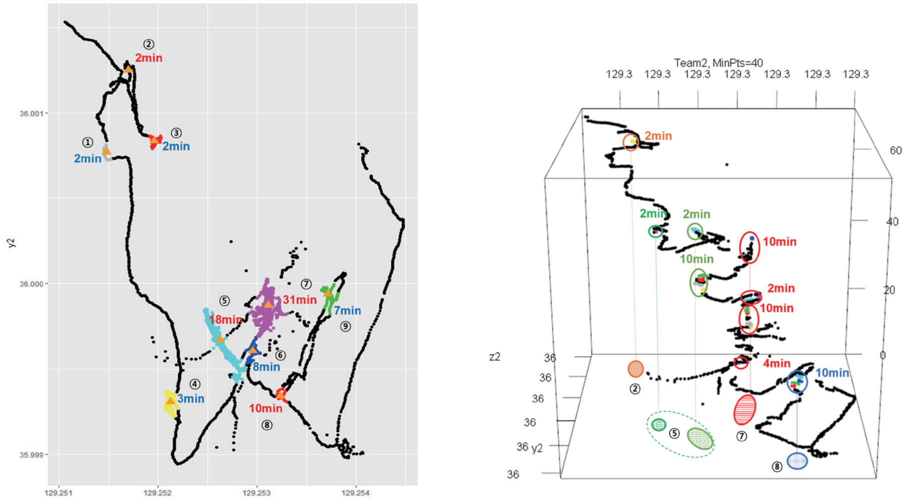
Table 7
Comparison of numbers of clusters with DBSCAN and 3D DBSCAN methods.

		T1	T2	T3	T4	T5	T6	T7	T8	Total
DBSCAN		27	9	8	9	19	10	14	4	100
3D DBSCAN	2D view	12	4	6	5	12	6	11	3	60
	3D view	22	9	7	7	15	8	13	4	85

Table 8
Cluster analysis in detail for Team 2.

Cluster #	2D DBSCAN (time and activities)		3D DBSCAN (time and activities)	
1	2 min	Observational activities	—	Disappeared
2	2 min	Observational activities	2 min	Observational activities
3	2 min	Observational activities	—	Disappeared
4	3 min	Observational activities	—	Disappeared
5	18 min	Observational activities	10 min	Observational activities
			2 min	Observational activities
			2 min	Investigating while walking slowly
6	8 min	Other activities	—	Disappeared
7	31 min	Other activities	4 min	Investigating while walking slowly
			10 min	Ice cream shop, resting
			2 min	Investigating surrounding buildings
			10 min	Ice cream shop, resting
8	10 min	Other activities	10 min	Checking the method of entering answers in the smart pad
9	7 min	Observational activities	—	Disappeared

In the case of Team 2, nine clusters are extracted from DBSCAN. However, when the cluster is analyzed using the 3D DBSCAN method, only four clusters remain in the 2D plan view and the other five clusters have disappeared. Moreover, the remaining four clusters are further subdivided into nine clusters when the time dimension is taken into consideration. Table 8 shows the cluster analysis results for Team 2 in detail. Five of the nine clusters extracted from DBSCAN are removed by 3D DBSCAN, namely, the 1st, 3rd, 4th, 6th, and 9th clusters. The 2nd and 8th clusters remain the same whether applying the DBSCAN or 3D DBSCAN method. However, the 5th and 7th clusters are subdivided into several clusters. For the 5th cluster, the 18 min spent as found by DBSCAN is subdivided into 10, 2, and 2 min spent in separate clusters, reducing the total staying time in the 5th cluster from 18 to 14 min when the 3D DBSCAN method is applied. For the 7th cluster, the 31 min spent according to DBSCAN is subdivided into 4, 10, 2, and 10 min, for a total in-cluster time of 26 min with 3D DBSCAN. Figure 10 shows the clusters of Team 2 in (a) 2D and (b) 3D spaces. Figure 10(a) shows nine clusters according to DBSCAN results, while Fig. 10(b) shows four clusters subdivided into nine clusters according to 3D DBSCAN results. These results indicate that the 3D DBSCAN method is more accurate than the DBSCAN method in the cluster analysis of spatiotemporal data.



(a) (b)

Fig. 10. (Color online) Cluster visualization of Team 2: (a) DBSCAN and (b) 3D DBSCAN.

4. Conclusion

In this study, we visualized and analyzed the characteristics of moving and staying patterns with spatiotemporal log data. As a case study, we collected and processed GPS log data generated by students who participated in inquiry-based fieldwork. The results of this study are as follows: First, we identified the spatiotemporal characteristics of the movement logs and staying patterns through STP visualization. Second, we investigated the spatial ranges of and time spent during the field activities by DBSCAN to identify staying places, namely, clusters. Third, we found that some clusters derived by DBSCAN are not true clusters, while the time spent in other clusters was overestimated when we investigated the time spent in each cluster more closely. To resolve this, 3D DBSCAN was applied in our study to find precise clusters. The results showed that the 3D DBSCAN method is effective in finding clusters of spatiotemporal data.

The results of this study showed that the 3D DBSCAN method is more effective than the DBSCAN method in the staying place analysis of spatiotemporal data. However, one limitation is that data analysis was carried out in the limited context of inquiry-based fieldwork activities over a 150 min period. In the future, the 3D DBSCAN methodology proposed in this study can be applied effectively in broader contexts of movement data analysis such as tourist travel patterns through SNS, trajectories of cars, vessels, or wildlife, and the movement of visitors in parks.

Acknowledgments

This work was supported by a National Research Foundation of Korea Grant funded by the Korean Government (NRF-2017S1A5B6066963).

References

1 J. Zhao, P. Forer, and A. S. Harvey: Inf. Vis. **7** (2008) 198. https://doi.org/10.1057/PALGRAVE.IVS.9500184

2 J. Zhao, P. Forer, Q. Sun, and D. Simmons: Cartogr. Geogr. Inf. Sci. **40** (2013) 349. https://doi.org/10.1080/15230406.2013.783449

3 P. O. Kristensson, N. Dahlback, D. Anundi, M. Bjornstad, H. Gillberg, J. Haraldsson, I. Martensson, M. Nordvall, and J. Stahl: IEEE Trans. Vis. Comput. Graph. **15** (2009) 696. https://doi.org/10.1109/TVCG.2008.194

4 F. O. Ostermann: Comput. Environ. Urban Syst. **34** (2010) 452. https://doi.org/10.1016/j.compenvurbsys.2010.05.007

5 T. M. T Do and D. Gatica-Perez: IEEE Trans. Mob. Comput. **13** (2014) 638. https://doi.org/10.1109/TMC.2013.19

6 R. Montoliu, J. Blom, and D. Gatica-Perez: Multimedia Tools Appl. **62** (2013) 179. https://link.springer.com/article/10.1007/s11042-011-0982-z

7 A. Thomason, N. Griffiths, and V. Sanchez: J. Comput. System Sci. **82** (2016) 566. https://doi.org/10.1016/j.jcss.2015.10.005

8 J. H. Choi, H. J. Lee, and Y. T. Park: J. KIISE. Software Appl. **39** (2012) 315. http://www.dbpia.co.kr/journal/articleDetail?nodeId=NODE01834424&language=ko_KR

9 R. A Abbaspour, M. Shaeri, and A. Chehreghan: Spat. Inf. Res. **25** (2017) 491. https://link.springer.com/content/pdf/10.1007%2Fs41324-017-0115-5.pdf

10 V. W. Zheng, Y. Zheng, X. Xie, and Q. Yang: ACM Proc. 19th Int. Conf. World Wide Web (WWW, 2010) 1029. https://link.springer.com/content/pdf /10.1145/1772690.1772795

11 D. Ashbrook and T. Starner: 2002 IEEE Wearable Comput (ISWC, 2002) 101. https://ieeexplore.ieee.org/document/1167224

12 S. J. Lee: User Behavior Reasoning and Next Location Prediction Based on Life Log Mining. Doctoral Dissertation, The Graduate School of Seoul National University (2016). http://dcollection.snu.ac.kr/public_resource/pdf/000000133607_20190731161150.pdf

13 O. Masutani, H. Iwasaki, K. Tei, Y. Fukazawa, and S. Honiden: Proc. 14th World Congr. ITS (WITS, 2007) 1. https://www.researchgate.net/profile/Osamu_Masutani/publication/233754020_Real-time_POI_detection_and_rating_using_floating_car_data/links/09e4150b3374e013fe000000/Real-time-POI-detection-and-rating-using-floating-car-data.pdf

14 V. W. Zheng, Y. Zheng, X. Xie, and Q. Yang: Artif. Intell. **184** (2012) 17. https://doi.org/10.1016/j.artint.2012.02.002

15 S. Kleinschmidt, A. T. Murray, S. J. Rey, L. Anselin, and T. H. Grubesic: Spat. Inf. Res. **24** (2016) 181. https://doi.org/10.1007/s41324-016-0019-9

16 G. S. Bhunia1, N. A. Siddiqui, P. K. Shit, N. Chatterjee, and S. K. Sinha: Spat. Inf. Res. **24** (2016) 639. https://doi.org/10.1007/s41324-016-0061-7

17 H. P. Kriegel, P. Kröger, J. Sander, and A. Zimek: WIREs Data Min. Knowl. Discovery **1** (2011) 231. https://doi.org/10.1002/widm.30

18 K. Pritee and R. D. Garg: Spat. Inf. Res. **25** (2017) 825. https://doi.org/10.1007/s41324-017-0148-9

19 N. Cho and Y. Kang: Spat. Inf. Res. **25** (2017) 141. https://doi.org/10.1007/s41324-016-0079-x

20 N. Cho and Y. Kang: J. Korean Cartogr. **18** (2018) 93. http://db.koreascholar.com/article.aspx?code=347494

21 I. Vasiliev: Cartographica: Int. J. Geogr. Inf. Geovis. **34** (1997) 1. https://doi.org/10.3138/D357-234G-2M62-4373

22 M. J. Kraak: Proc. 21st Int. Cartogr. Conf. (ICC, 2003) 1988. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.102.9231&rep=rep1&type=pdf

23 N. Pelekis, I. Kopanakis, E. Kotsifakos, E. Frentzos, and Y. Theodoridis: 2009 9th IEEE Int. Conf. Data Mining (ICDM, 2009) 417. https://doi.org/10.1109/ICDM.2009.57

24 H. Michael, P. Matthew, and D. Derek: J. Stat. Software **25** (2017) 409. https://pdfs.semanticscholar.org/0c4c/3ddb6820cf59d29dc1dec7c71a4a6eecee27.pdf

About the Authors

Nahye Cho received her B.S. degree from Daegu University, South Korea, in 2011 and her M.S. and Ph.D. degrees from Ewha Womans University, South Korea, in 2013 and 2018, respectively. From 2018 to 2019, she was a research professor and a postdoctoral student at the University of Seoul, South Korea. Since 2019, she has held a postdoctoral position at Ewha Womans University. Her research interests are in spatiotemporal data visualization and spatial statistics. (cho.nahye@gmail.com)

Youngok Kang is a professor of social studies education and GIS convergence track at Ewha Womans University. From 2013 to 2014, she served as the chairman of the Korean Cartographic Association and is currently the chairman of the Korean Society for Geospatial Information Science. Her research areas are GIS, spatial data analysis, and social big data analysis. Her most recent research interest is in spatiotemporal big data analysis using deep learning technology. (ykang@ewha.ac.kr)