Filtering of Erroneous Positioning Data with Iterative Application of One Class Support Vector Machine †

Woojoong Kim and Ha Yoon Song
Department of Computer Engineering, Hongik University, Seoul, Korea
veritaswjkim@gmail.com, hayoon@hongik.ac.kr

Abstract

The topics on human mobility model have long been researched by various academic and industrial fields. It also has been proven that human mobility has specific patterns and can be predicted up to the probability of 93%, since the mobility of a person cannot be random while peoples have their own frequent visiting places. As a basis of human mobility research, sets of positioning data is used widely. The positioning data of a human can be obtained by GPS or other similar positioning systems, however, it contains inherited environmental errors. It is clear that such position errors harm the correctness of human mobility related results. In this paper we will present filtering method of erroneous positioning data of human mobility with the use of One-Class Support Vector Machine (OCSVM), and we adapted Radial Basis Function (RBF) as kernel function. Experimental values of the critical parameter for RBF have been found for optimal filtering. With this optimal parameter, we filtered raw data set of human mobility trail in order to obtain accurate position data set for further research purpose. By iteratively applying the OCSVM based filtering, like hill climbing approach, we prove that researchers can filter arbitrary rate of raw data for their own purpose. With four sets of positioning data set from various sources, we demonstrate the usefulness of our filtering approach.

Keywords: *Human Mobility, Global Positioning System, Positioning Data, Error Filtering, One Class Support Vector Machine, Radial Basis Function, Parameter Optimization, Hill Climbing*

1. Introduction

The resent advances of mobile devices enable various location based services over human mobility, especially the introduction of smart phone with GPS or other positioning equipment. However these positioning data have error according to its operational environment. In such cases, many of applications require filtering of such erroneous positioning data. As we experienced by our experiments, more than 12% of positioning data were erroneous by use of smart phones. This simple experiment was done by use of smart phone app over

† This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology(No. 2011-0025875). This paper is a revised and expanded version of a paper entitled "Optimization Conditions of OCSVM for Erroneous GPS Data Filtering" by Woojoong Kim and Ha Yoon Song and presented at "FGIT-MulGraB, Jeju, Korea, 8th-10th December 2011".

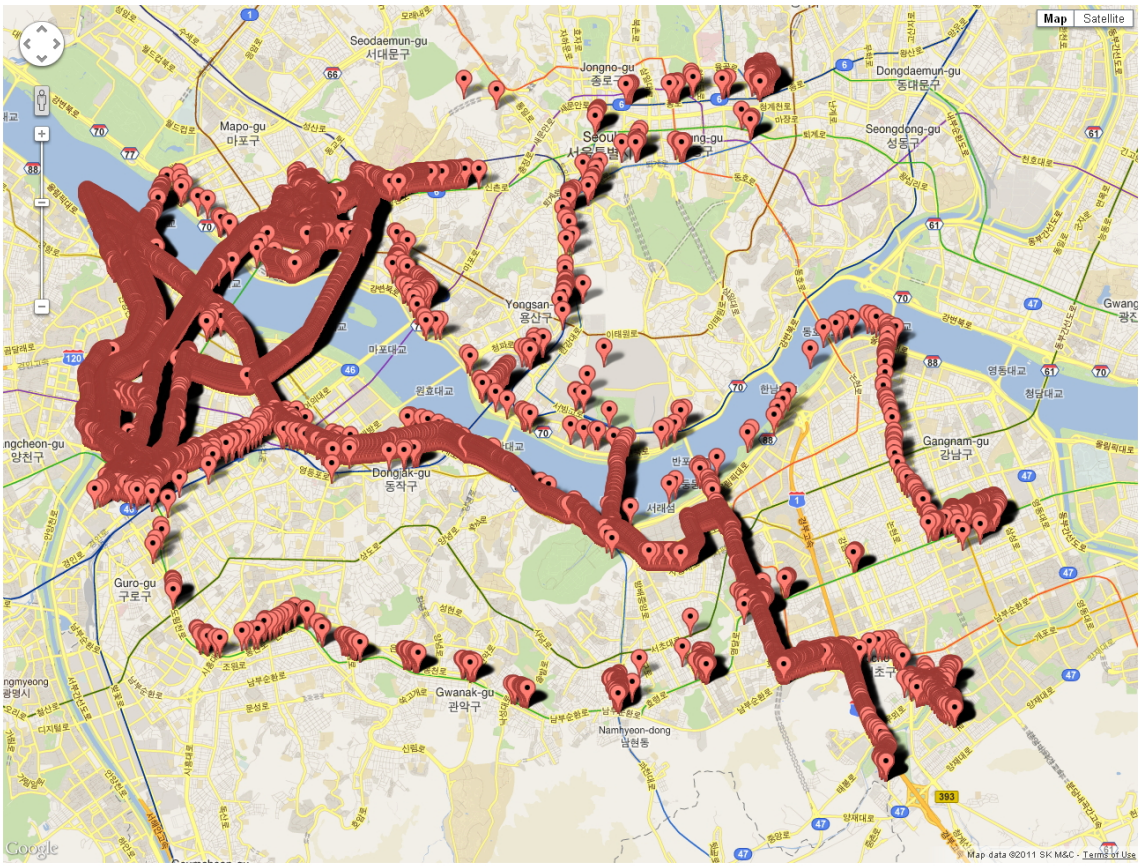

Figure 1. A Trail of iPhone Positioning Data Set over Seoul, Korea

Samsung Galaxy Tab and it internally uses the position of 3G base station and done by commercial GPS positioning device.

Another aspect of location based research is regarding the human mobility model. Several interesting results were drawn from psychologist. The results contain route selection of a person [1], transportation method selection [2][3][4], which denotes psychological factors are major factor of human mobility as well as functionality factors of human mobility [5]. Also habits of humans are major factors such as commuting habits concerned with transportation methods [6].

Another research field of complex system physics showed that up to 93% of human mobility can be predicted since peoples avoid the random selection of next destination instead selects their place frequented and their route frequented [7].

On the result of past research results, it is now possible to obtain positioning data of a human mobility pattern with GPS or other positioning systems as shown in figures 1, 2, 3 and 4. The sets of positioning data will be a basis for human mobility model construction.

In this paper, we will propose a filtering technique which filters erroneous positioning data with the use of One-Class Support Vector Machine (OCSVM) [8]. The Radial Basis Function (RBF) of OCSVM has a specific parameter called γ and it affects a trade-off between class size and density. The key point of using OCSVM is to figure out the appropriate range of γ. With the various sets of positioning data, we conducted experiments for optimal parameter value of RBF and then presented our experimental results. We introduced in-

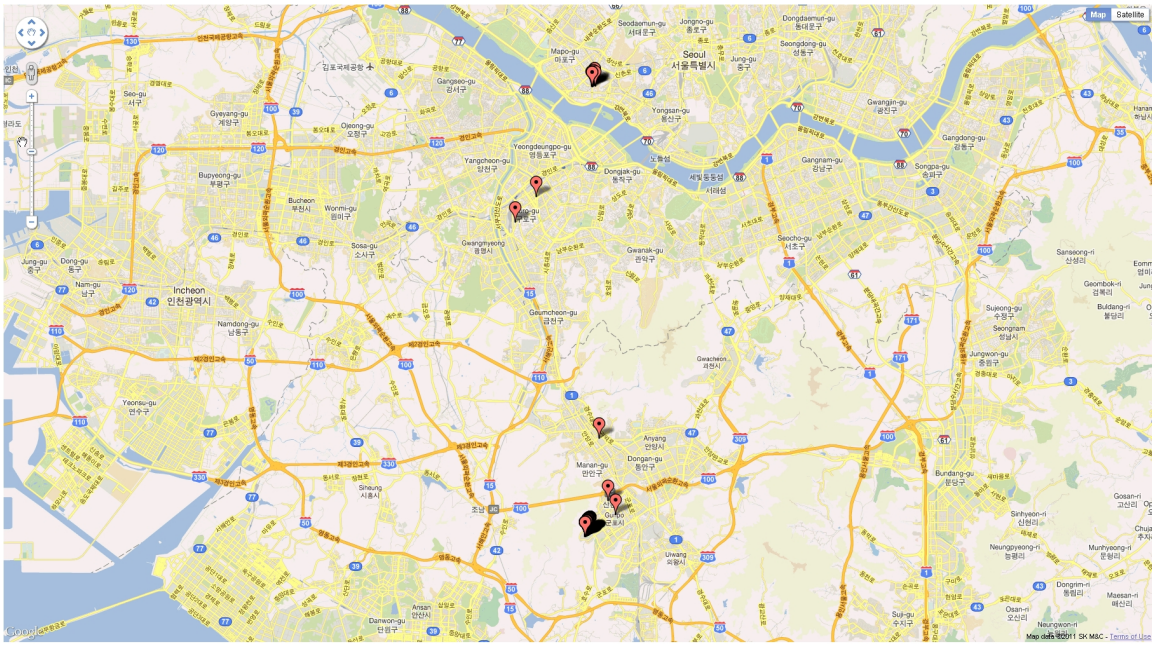

Figure 2. A Trail of 3GBS Positioning Data Set over Seoul, Korea

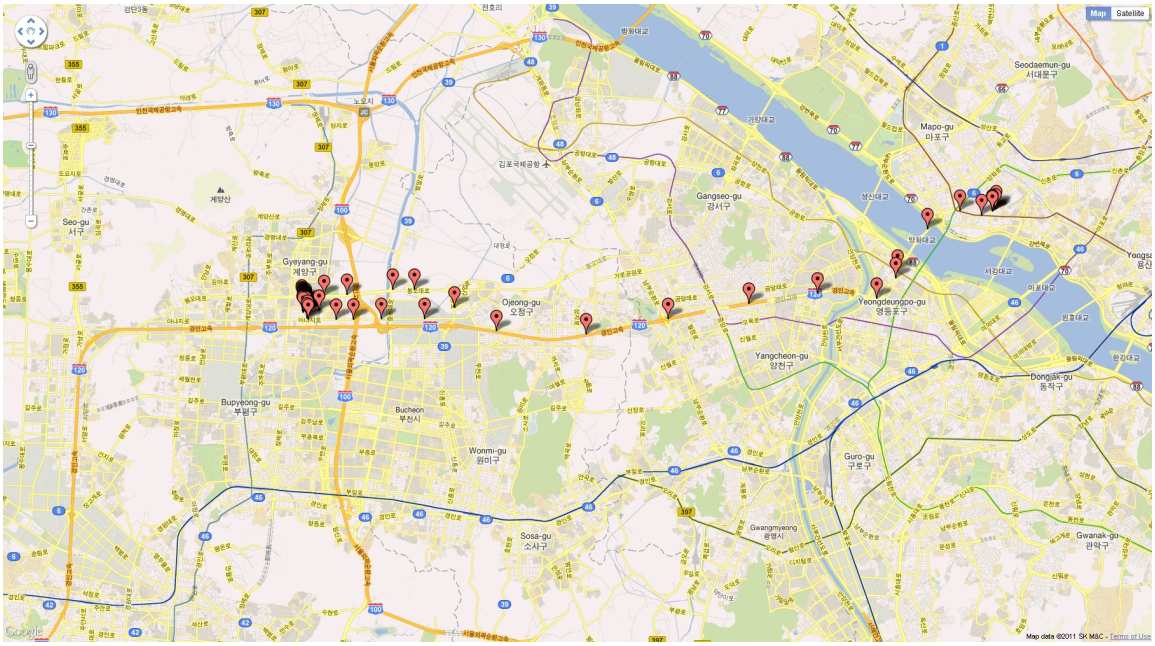

Figure 3. A Trail of GPS Positioning Data Set II over Seoul, Korea

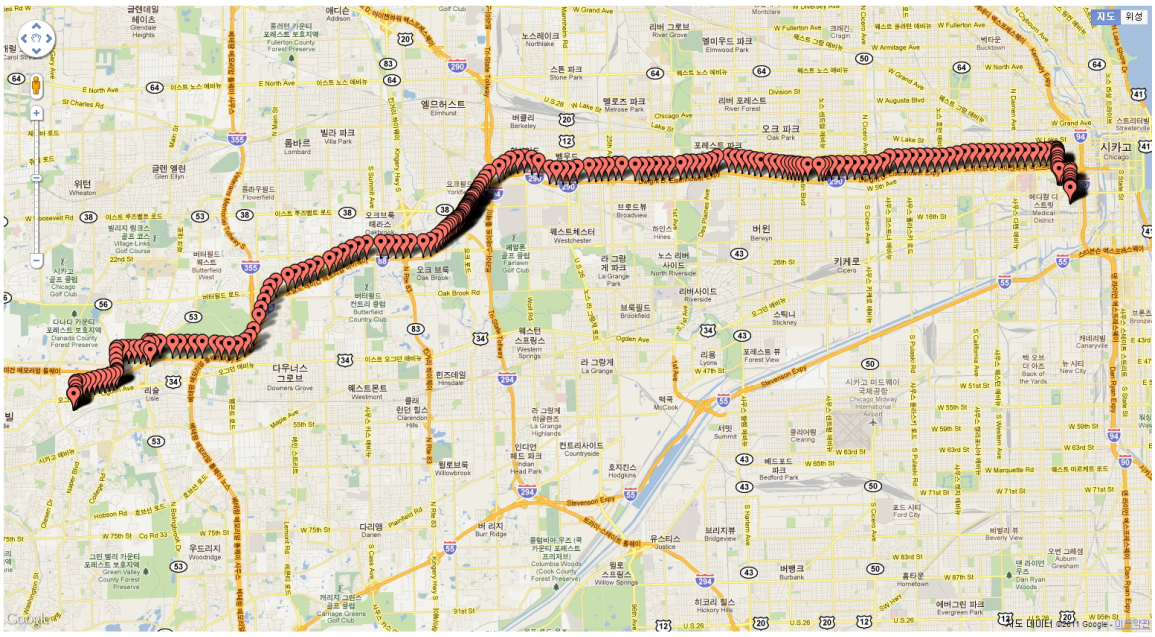

Figure 4. A Trail of GPS Positioning Data Set over Illinois, USA

cremental inclusion of positioning data for proper filtering mechanism since OCSVM tends to filter about 50% of input data due to the nature of OCSVM. Therefore, we must have filtering mechanism that controls ratio of input data filtering arbitrarily.

In section 2 we summarizes the OCSVM, and the following section 3 shows our technique for filtering, and basic experiments for RBF parameter calibration. Section 4 shows our experimental results. The results will be discussed for optimal RBF parameter in section 5. Basic filtering results will be shown in section 6 with the enhancement of filtering mechanism using incremental iteration of filtering inputs. We will finalize this paper in section 7.

2. One Class Support Vector Machine

We will discuss about OCSVM(One-Class Support Vector Machine) and its kernel function, RBF(Radial Basis Function). OCSVM is one kind of SVM(Support Vector Machine). SVM divide inputs into two classes, i.e. maps inputs onto $\{-1, 1\}$. SVM is usually used for pattern recognition such as face recognition [9]. And it used for classification of raw data, for example SVM classifier for cork stopper classification can be found in [10].
OCSVM was proposed in 2001 and it filters input data into two categories: class members and others.
For a set of data with the cardinality of N,

$$X_1, X_2, \ldots, X_n \in \aleph, n \in N$$

denotes members of the data set. These data elements will be placed into *Feature Space* in higher dimension than the dimension of data elements. Thus we can lead an optimality problem which minimizes the size of *Hyper Sphere* with maximum number of data elements. We must use a kernel function in order to increase the dimension as Radial Basis Function

shown in equation 1.

$$\Phi(u, v) = e^{(-\gamma \times |u-v|)} \tag{1}$$

Where u and v are coordinates of data elements on the space and $|u - v|$ stands for their *Norm value*. Here comes the parameter value of RBF, γ, which is user controllable for his or her specific use.

3. Experimental Process

3.1. Basic Filtering Algorithm

We basically use sets of positioning data, representing a mobility trail of a human. The positioning data is usually in a form of <*latitude*, *longitude*, *time*, *id* >where *id* is identification of a set, and they compose sets of time sequenced data. This positioning data set collected by the authors will be also used for primary filtering experiments.
With the every two consecutive pair <$latitude_2$, $longitude_2$, $time_2$, *id* >and <$latitude_1$, $longitude_1$, $time_1$, *id* >in data set, we can calculate the difference of latitude and longitude as shown in equations 2 and 3.

$$\Delta Latitude = \frac{Latitude_2 - Latitude_1}{t_2 - t_1} \tag{2}$$

$$\Delta Longitude = \frac{Longitude_2 - Longitude_1}{t_2 - t_1} \tag{3}$$

Then we can get a coordinate tuple of

$$(\Delta Latitude, \Delta Longitude) \tag{4}$$

which is a very small value and normalized by multiplying arbitrary constant such as 300,000 in our experiments. Those set of tuples, named as *points*, can be drawn as shown in figure 5. Set of data will be used as input for OCSVM with various RBF parameters. We call the resulting values as *pixels* which compose set of total class.
Also, number of points in a class, number of total points, and number of slow points are notable parts of results. The slow points of human mobility are tuples having speed value of less than 5Km/h, while other tuples will be regarded as mobile points.
The filtering process is divided into three categories like the followings:

- Filtering only with slow points: the RBF parameter with slow points will be bases as we did in subsection 3.2.
- Filtering only with mobile points: the number of mobile points are dominating and will greatly affect the RBF parameter value.
- Filtering with both slow and mobile points.

Several resulting values can be drawn from the result of filtering process, such as size of cluster, density of cluster, and hit rate of points. Density can be drawn from the equation 5:

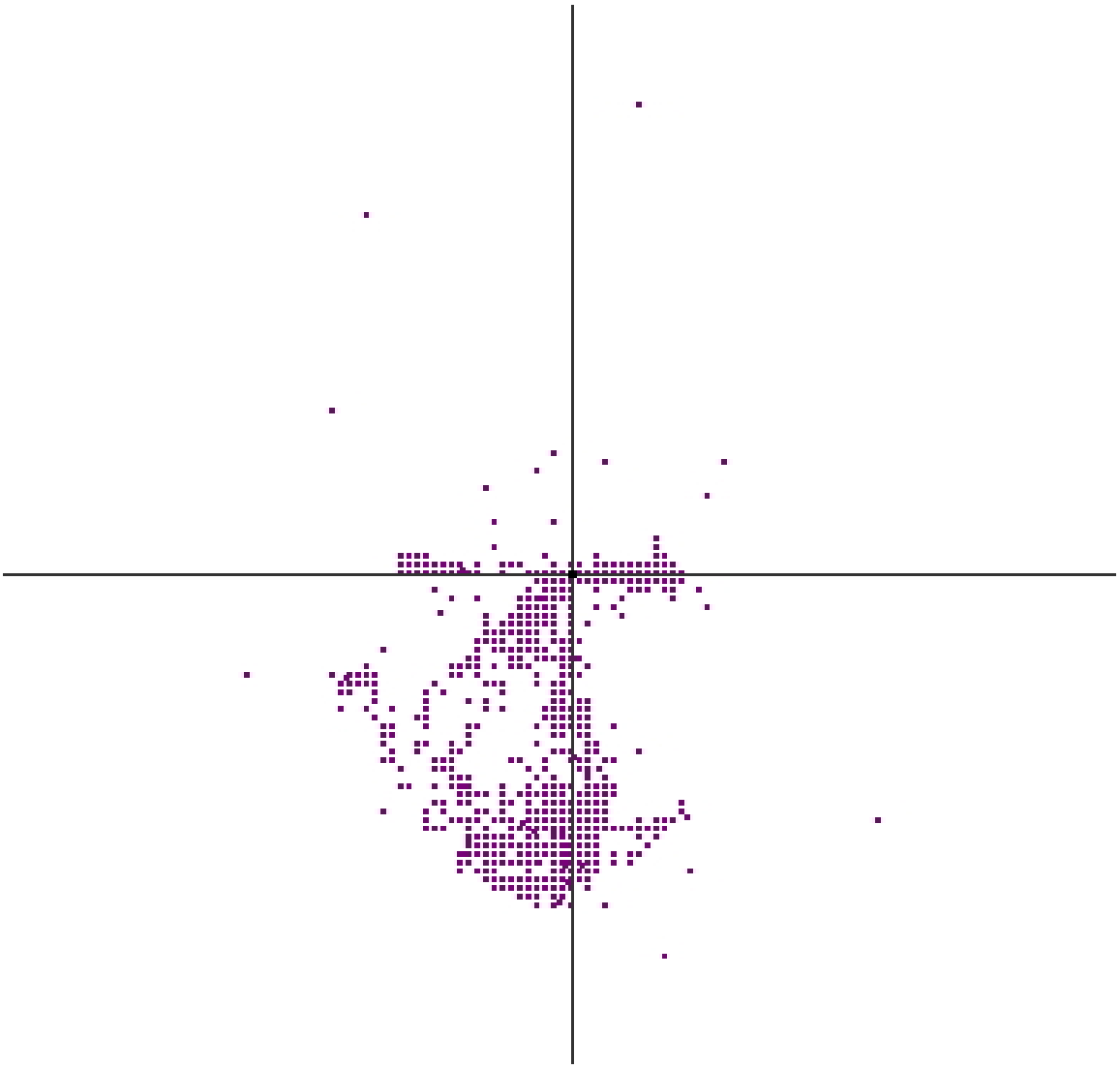

Figure 5. A Typical Representation of Data Set

$$Density = \frac{n(P_{Class})}{n(Class\,Size)} \tag{5}$$

where $n(P_{Class})$ is number of points in a class and $n(Class\,Size)$ is number of pixels in a class. And the hit rate of class can be defined as equation 6:

$$Hit\,Rate = \frac{n(P_{Class})}{n(P_{Total})} \tag{6}$$

where $n(P_{Class})$ is number of points in a class and $n(P_{Total})$ is number of points as inputs, i.e. total number of points. For position error filtering purpose, we had implemented OCSVM filter using LIBSVM(Library Support Vector Machine) for GPS anomaly detection. An example of libSVM based work can be found in [11].

Table 1. Result of Base Experiments

Building	*3G Base Station GPS*		*GPS*	
Inside	n(Data Point)	893	n(Data Point)	2186
	n(Error Point)	434	n(Error Point)	939
	Error Rate	48.6%	Error Rate	43.0%
	E[Error Dist]	52.5530m	E[Error Dist]	43.5506m
	Max(Error Dist)	156.7578m	Max(Error Dist)	10769.72m
	$\sigma_{ErrorDist}$	32.6859m	$\sigma_{ErrorDist}$	370.6034m
	Range of γ	$5 \times 10^{-3} \sim 5 \times 10^{-6}$	Range of γ	$5 \times 10^{-3} \sim 5 \times 10^{-5}$
Outside	n(Data Point)	331	n(Data Point)	1690
	n(Error Point)	122	n(Error Point)	208
	Error Rate	36.9%	Error Rate	12.3%
	E[Error Dist]	52.6618m	E[Error Dist]	4.4498m
	Max(Error Dist)	206.3526m	Max(Error Dist)	51.7789m
	$\sigma_{ErrorDist}$	23.5953m	$\sigma_{ErrorDist}$	7.1696m
	Range of γ	$5 \times 10^{-3} \sim 5 \times 10^{-5}$	Range of γ	$5 \times 10^{-3} \sim 5 \times 10^{-5}$

3.2. Basic Experiment for Calibration

We experienced a basic test as our base experiment to check the positioning data accuracy and calibrate the initial RBF parameter value. We fix positioning devices both outside area and inside the building, and collected positioning data for several hours without moving any device. The first positioning device is Garmin GPSMAP62s [12] for pure GPS data collecting. The second positioning device is Samsung Galaxy Tab to obtain positioning data from its connected 3G base stations (3GBS). We guess Galaxy tab will show more error in both situation, and both of the data set from GPS and 3GBS shows positioning error, especially inside the building. The result of this basic experiment is listed in table 1.

The variance in position data is regarded as errors and distance of error can be calculated form the position data. As we guessed, 3GBS shows larger error rate, larger error in distance, larger maximum error distance, and bigger standard deviation in error distance. Due to the producer's policy of Garmin GPSMAP62s, which estimates the user's location upon past velocity while it lost the GPS signal, it shows drastic error value inside the building. Thus we think the GPS inside a building cannot be a meaningful data. GPS data from outside area is very accurate enough for precise localization and even the maximum error distance is in a reasonable range of 52 meters.

We concluded that the following experiments with real human mobility data will be based on the positioning data sets of GPS and 3GBS from outside area. The value range of RBF parameter (γ) shown in table 1 will be considered as a core parameter value for further experiments.

4. Experimental Results

We use the following sets of positioning data for our experiments.

- iPhone: positioning data collected by authors iPhone 3GS over Seoul, Korea, in (2011.OCT.31 $\sim$ 2011.NOV.30) and is shown in figure 1.

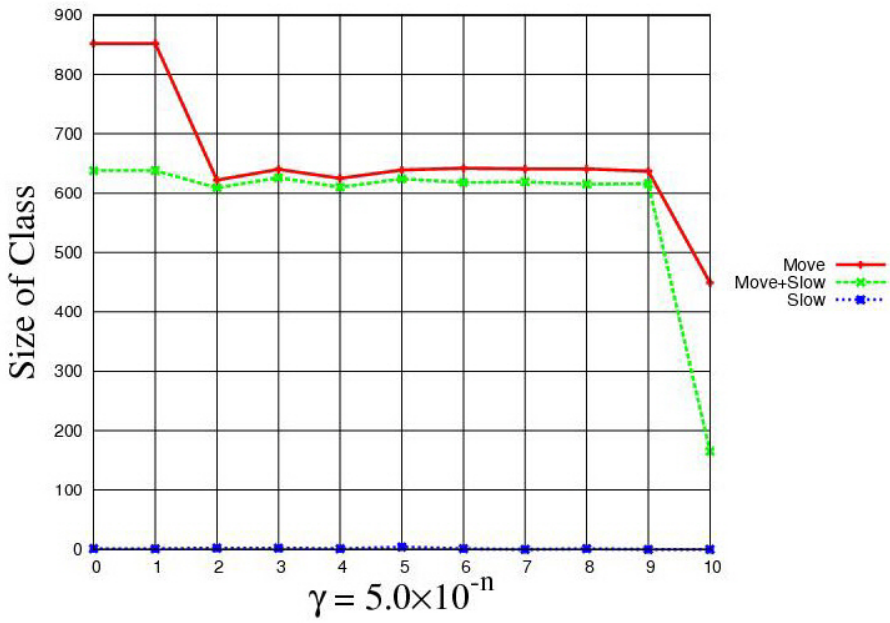

Figure 6. Size of Class over iPhone data set

- 3GBS: 3G cellular base station positioning data over Metropolitan Seoul Area, Korea, voluntarily collected in (2011.JUN.03 $\sim$ 2011.JUN.06) by one author of this paper and is shown in figure 2.
- GPS-Seoul: GPS data over Metropolitan Seoul Area, Korea, voluntarily collected in (2011.JUL.18 $\sim$ 2011.JUL.26) by a researcher and is shown in figure 3.
- GPS-Illinois: GPS data over Illinois, USA collected in (2006.MAY.30 $\sim$ 2006.MAY.31) as shown in [13] and is shown in figure 4.

We will focus on the iPhone data set in this paper. The iPhone data set is collected by porting data collecting app on iPhone 3GS with iOS5. The iOS5 decides current position with the mixture of crowd source WIFI positioning, cellular station location, and GPS [14]. The app is designed to record positioning data whenever it senses a location change of iPhone or for every 3 second in immobile state. Of course, we can change the period of data collection for immobile state from 3 second to 60 second.
Figures 6, 7, 8 shows the result for iPhone data set. Figures 9, 10, 11 shows the result for GPS-Illinois data set. As well figures 12, 13, 14 shows similar results for the 3GBS data set.

The data legends of Slow denotes result for slow points only, Move denotes results for mobile points only and Move+Slow denotes for total points. Slow points represent points with less than 5Km/h of speed. In our filtering strategy, slow points can be regarded as pivot values. For every figure, the x-axis stands for n of $\gamma = 5 \times 10^{-n}$. *Size of Class* stands for the size of class in pixels, and *Density of Class* can be calculated as equation 5. For example, density value of 100 means that 100 points are mapped onto one pixel. Figure with *Hit Rate* shows the hit rate as denoted in equation 6. We experienced big fluctuation for slow points in figures 6, 7, and 8 however it is negligible since the iPhone data set has

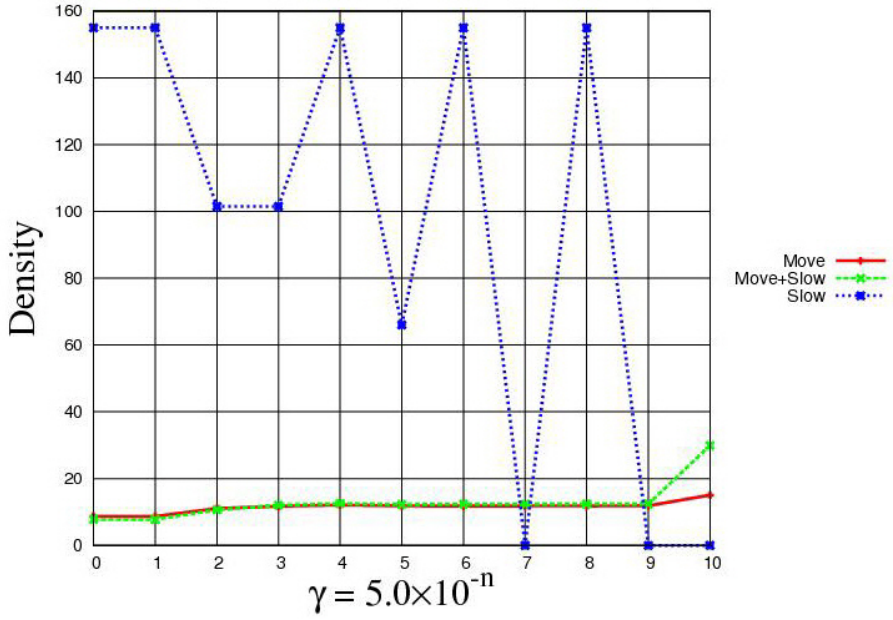

Figure 7. Density of Class over iPhone data set

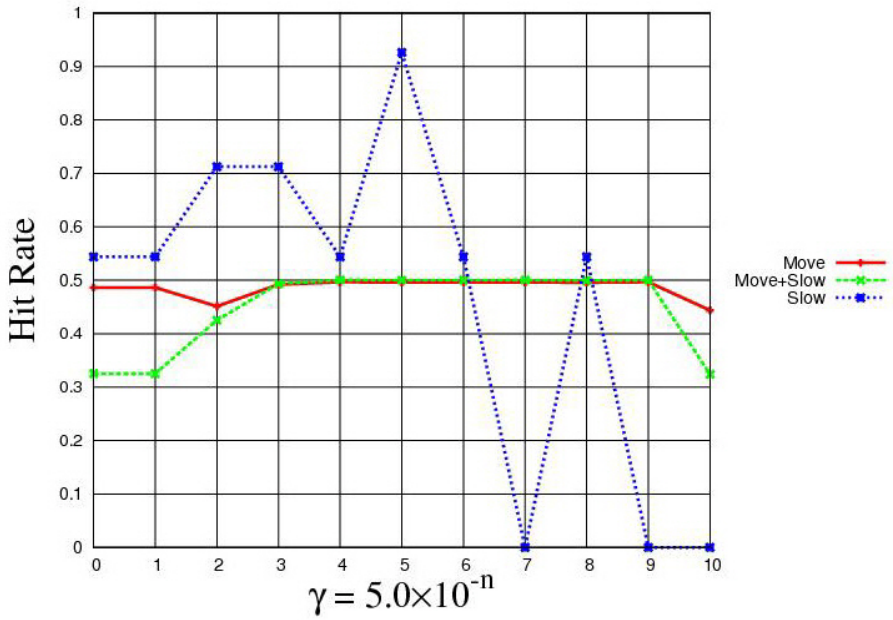

Figure 8. Hit Rate of Class over iPhone data set

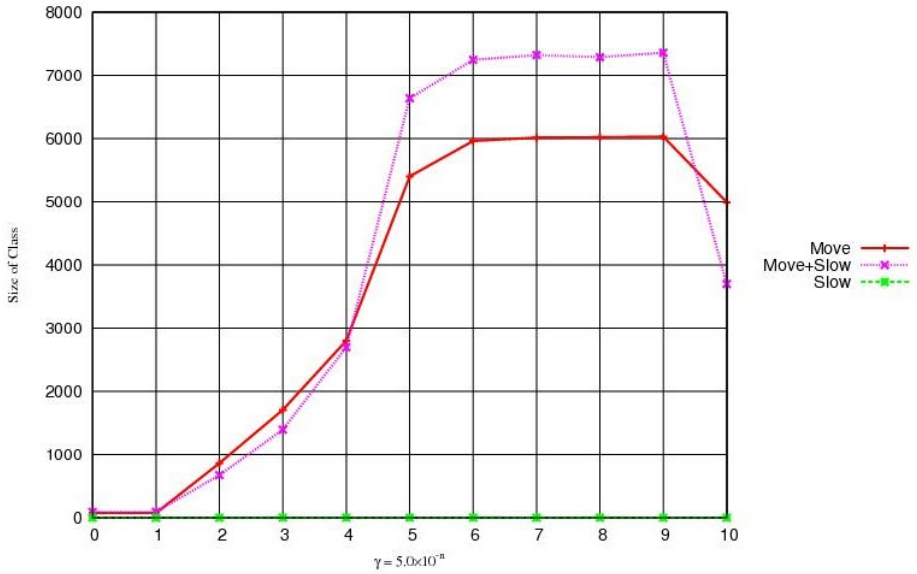

Figure 9. Size of Class over GPS data set

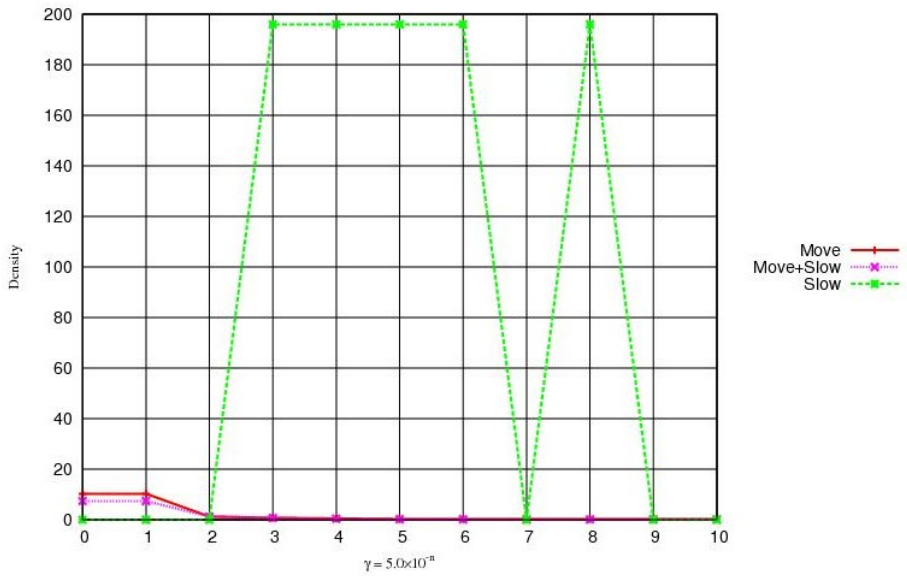

Figure 10. Density of Class over GPS data set

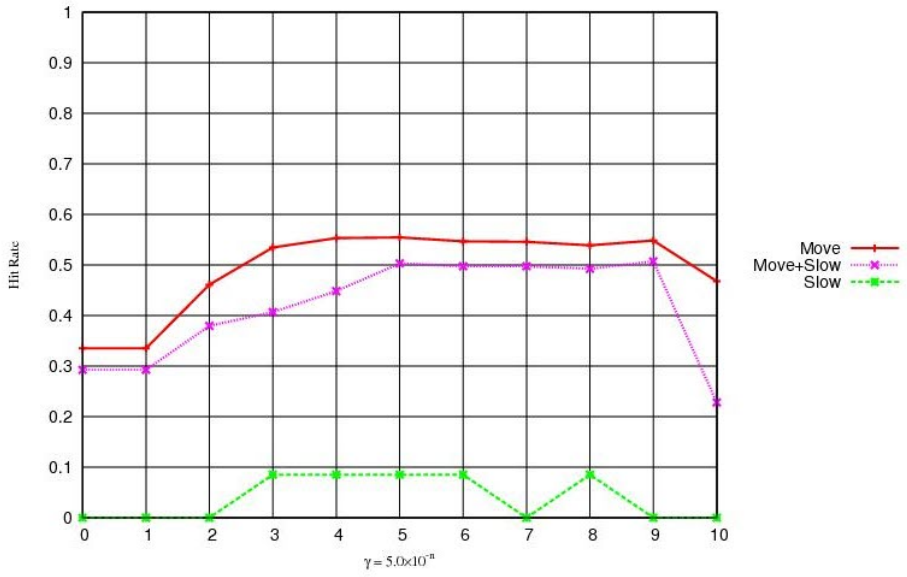

Figure 11. Hit Rate of Class over GPS data set

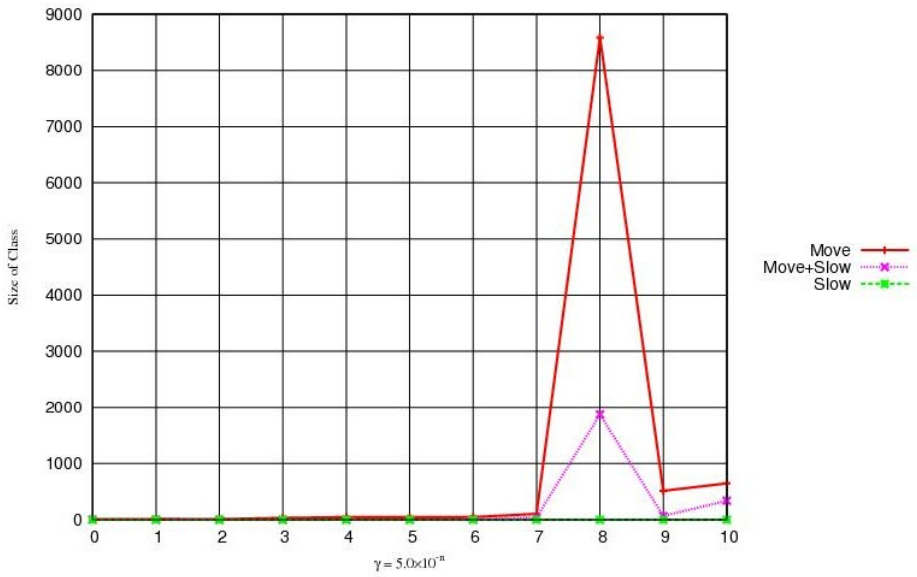

Figure 12. Size of Class over 3GBS data set

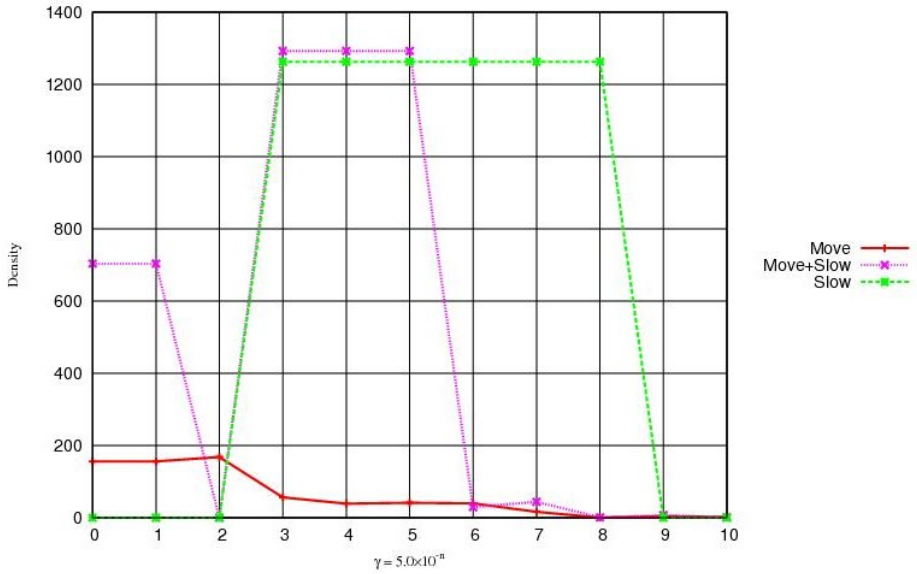

Figure 13. Density of Class over 3GBS data set

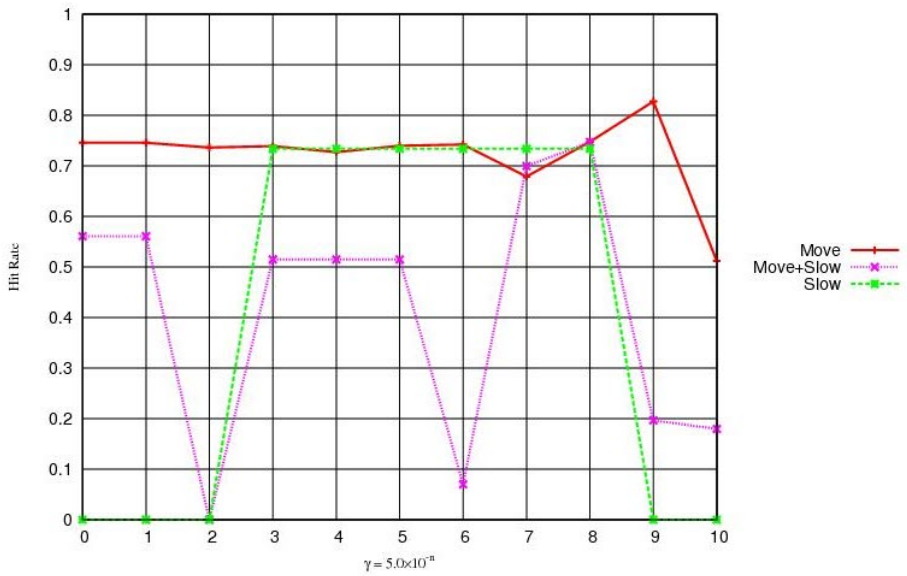

Figure 14. Hit Rate of Class over 3GBS data set

less than 300 slow points out of 15,251 points, This phenomenon is due to the positioning data collection mechanism which eagerly collects points whenever it senses the mobility of iPhone. Also, the collection mechanism collects data whenever it senses false data, i.e., error position. In case of iPhone data set, set of slow points cannot be regarded as pivots.

We also experienced stability in class size and number of class members with RBF parameter γ more than $5.00 \times 10^{+01}$, where the class size is 858 and 8588 data in the class for the case of iPhone. We noticed almost exact similarities between GPS-Illinois data set and GPS-Seoul data set therefore we omit the result for GPS-Seoul data set.

5. The Parameter for Radial Bases Function

In order to figure out proper RBF parameter, we must consider three results.

For size of class, density of class, and hit rate, there are range of γ being stable. And the intersection of range set for the three results will be proper range of γ. The consideration must be taken care for Slow data, Move data, and Move+Slow data. In case there tends to be contradiction between Move data set and Slow data set, dominating sets in cardinality will be a considerable one. Similar considerations can be taken in case of contradictory situation among size of class, density of class, and hit rate. In such cases, we must consider hit rate and density of class as major ones since size of class is just a counted values while two others are calculated rates.

As the numerical results shows in section 3, we found the value of γ parameter in $(0, 5.00 \times 10^{-01}]$ is meaningless because of the nature of RBF as shown in equation 1. From table 1, we already decided that the range of γ must be in $[5.00 \times 10^{-03}, 5.00 \times 10^{-05}]$.

For iPhone data set, we found that the range of γ is in $[5.00 \times 10^{-03}, 5.00 \times 10^{-06}]$. The parameter value in $[5.00 \times 10^{-05}, 5.00 \times 10^{-09}]$ are stable in GPS-Illinois data sets with both Move data and Move+Slow data. While the Slow data from GPS-Illinois set shows RBF parameter value in $[5.00 \times 10^{-05}, 5.00 \times 10^{-06}]$. The range of γ for GPS-Seoul data set is very similar to that of GPS-Illinois. As well, 3GBS data set shows that the parameter value in $[5.00 \times 10^{-03}, 5.00 \times 10^{-05}]$ are stable.

Therefore, we can conclude that the RBF value $\gamma = 5.00 \times 10^{-05}$, which is intersection of all appropriate RBF parameter range, is somewhat meaningful for both GPS and 3GBS positioning data filtering.

As well as exponential variance in γ values, we conducted linear variance in RBF parameters as one of minor experiments. We experienced very small difference in this linear case, and only trade-offs between class size and class density can be found. We also experienced stability in class size and number of class members with RBF parameters over than $\gamma = 5.00 \times 10^{+01}$, where the class size is 120 and 1176 data in the class.

6. Iterative Incremental Filtering

With the RBF value $\gamma = 5.00 \times 10^{-05}$, we did initial filtering for positioning data sets.

By the help of Google map, we can visualize the result as set of points on a geographical map. Figure 15 is a visualization of filtering result for iPhone data set, figure 16 is a visualization of filtering result for 3GBS data set, figure 17 is a visualization of filtering result for GPS-Seoul data set, and figure 18 is a visualization of filtering result for GPS-Illinois data set.

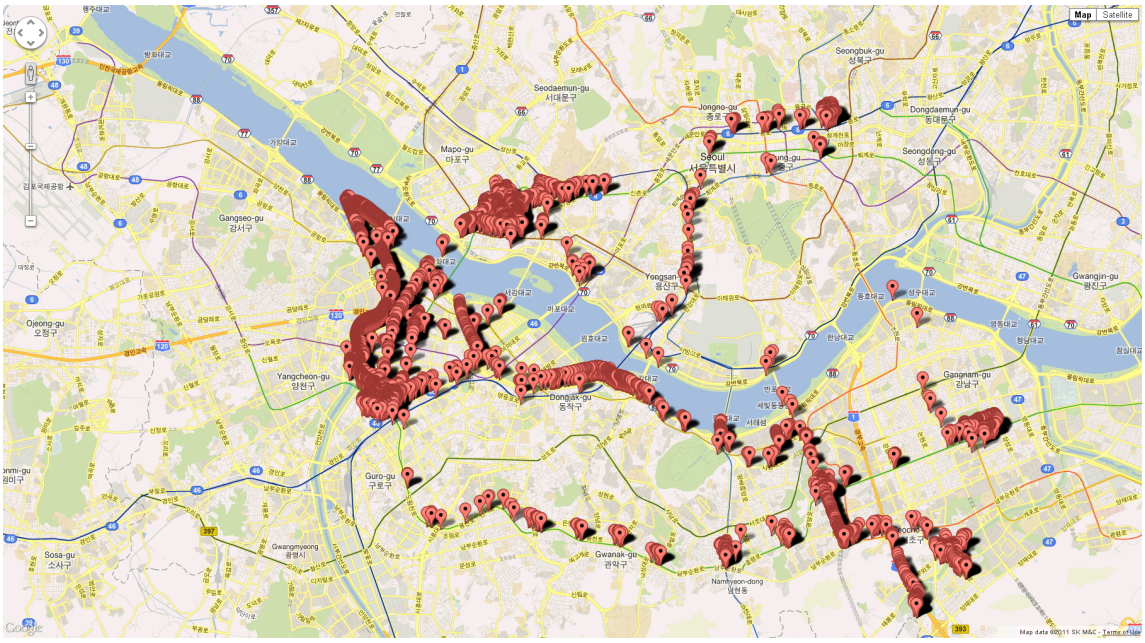

Figure 15. Initial Filtering result of iPhone Positioning Data Set

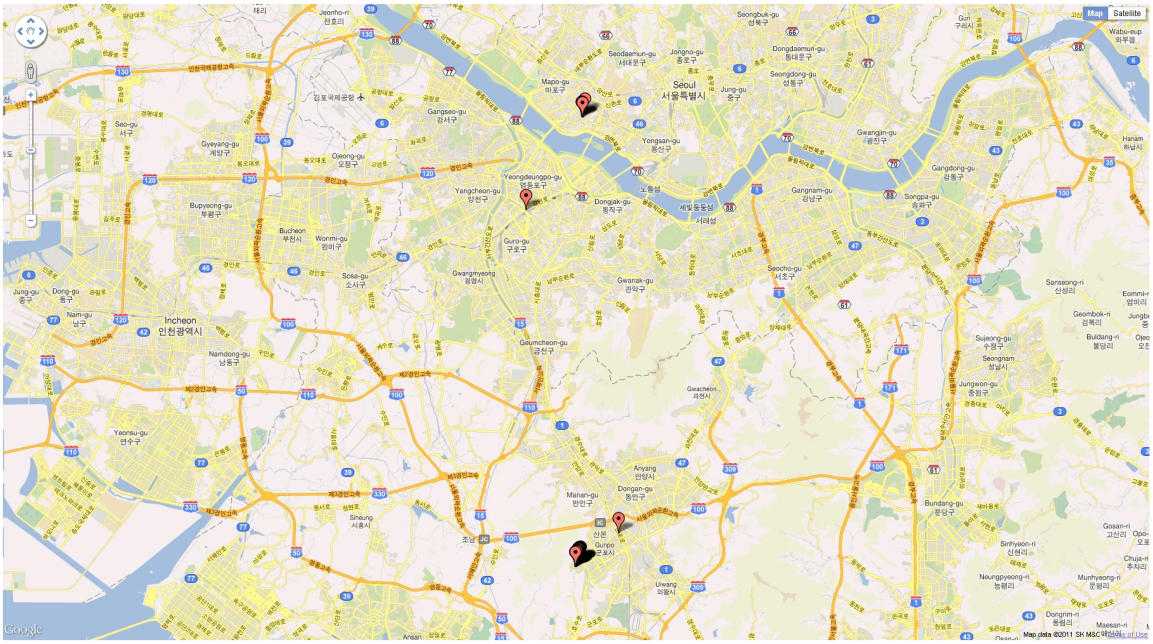

Figure 16. Mapping of Filtered Positioning Data from 3GBS Data Set

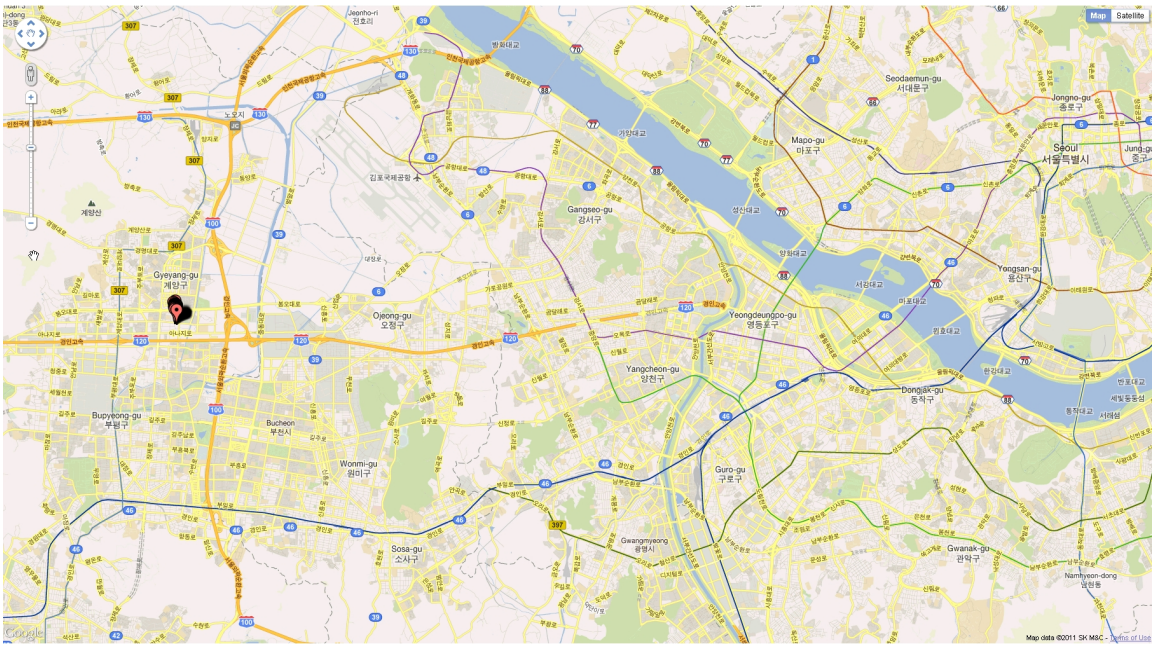

Figure 17. Mapping of Filtered Positioning Data from GPS-Seoul Set

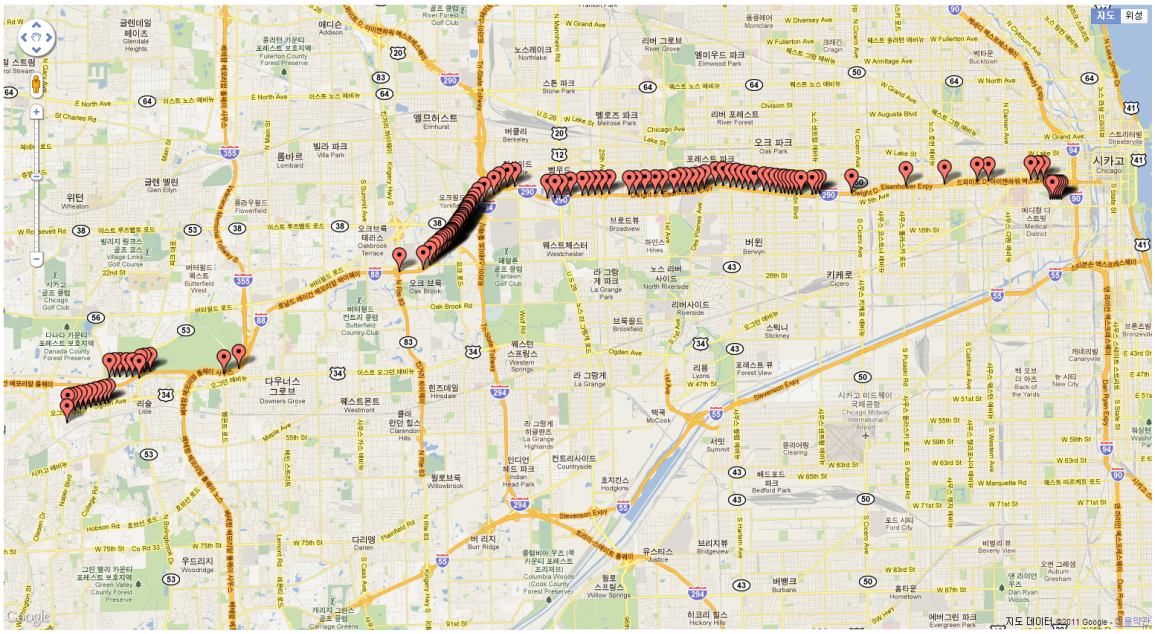

Figure 18. Mapping of Filtered Positioning Data from GPS-Illinois Positioning Data Set

Table 2. Result of Iterative Filtering with 15251 points

Iteration Count	1	2	3	4	5	6
Number of Input Points	15251	7624	3809	1901	949	475
Number of Filtered-in Points	7625	3813	1904	949	472	237
Number of Filtered-out Points	7624	3809	1901	949	475	237
Cumulated Filtered-in Points	7625	11438	13342	14291	14763	15000
Hit Rate	0.500	0.750	0.875	0.937	0.968	0.984
Reference Figure No.	15	19	20	21	22	23

Investigating this initial result, it shows the tendency of over-filtering, i.e. correct data could be filtered-out. Due to the nature of OCSVM, our mechanism filters out about 50% of data. In other words, OVSCM maps input data onto $\{1, -1\}$ with the same rate.

In order to overcome the over-filtering phenomena, we introduced hill climbing approach, iterative incremental inclusion of filtered-out data. For each iteration, the filtered-out data set can be isolated and re-filtered by the algorithm iteratively and added to the filter-in result of the previous step. In sum, the iterative incremental filtering can be implemented as follows:

1. N is number of desired iteration.
2. Name the input data as S^0_{out}.
3. Set iteration count n as 0.
4. Do OCSVM filtering on S^n_{out}.
5. Set S^{n+1}_{in} with filtered-in data and set S^{n+1}_{out} with filtered-out data.
6. Increment n by 1.
7. if n < N goto 4.
8. Output $S_{in} = \cup_{i=1}^{N} S^i_{in}$.

where S_{in} stands for filtered-in data after iterative incremental filtering.

We found this mechanism is similar to hill climbing method [15] without a problem of local maxima due to the nature of our filtering problem.

Table 2 shows the result of our approach for iPhone set. We focused on iPhone data set since iOS5 has every possible positioning mechanism and thus every nature of positioning error. In addition, it contains sufficient number of data for iterative incremental approach since it had been collected over the longest duration among out positioning data sets. Also, it covers the widest area among our data sets.

With more than five iterations, more than 96.8% of data will be included and thus the filtering result will be meaninglessly saturated.
We conclude that our filtering algorithm shows appropriate result with 2 to 4 steps of iteration for filtering considering the usual error rate in table 1. For example, we can conclude three iterations are enough to include 87.5% of positioning data.

We can filter positioning data with arbitrary filtering rate with an idea of bisectioning. Intermediate filtering results can also be filtered again for desired filtering rate. For example, if we apply OCSVM based filtering step for S^1_{in} instead of S^2_{out}, the union of $S^1_{in} \cup S^3_{in}$ will filter 62.5% if input data.

Figure 19 shows the result of second iteration for iPhone data set. As aforementioned,

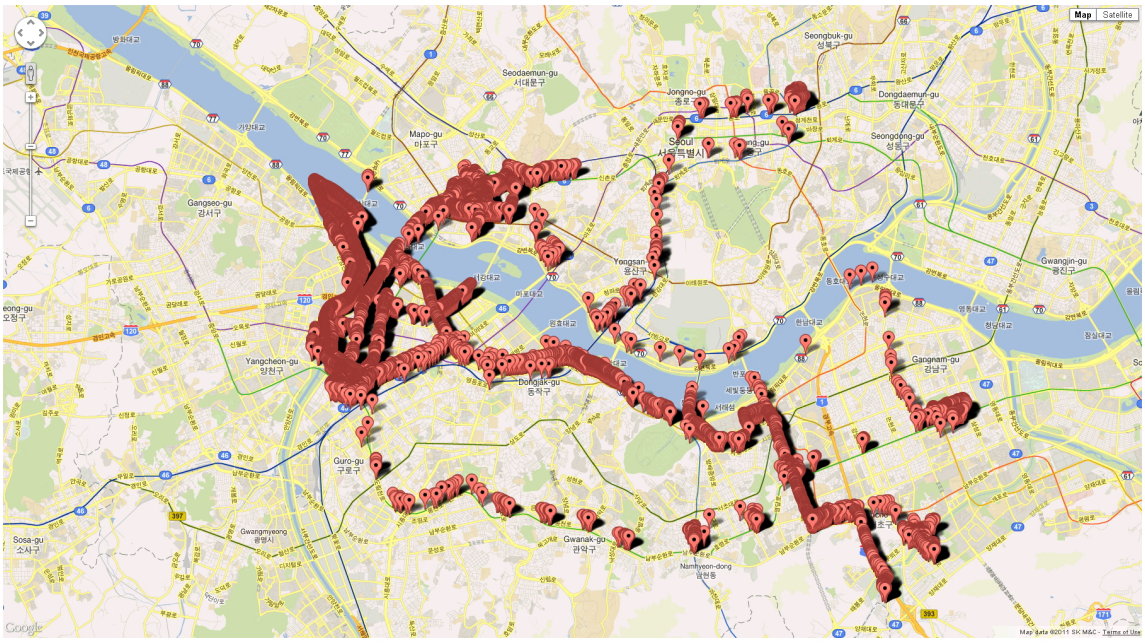

Figure 19. Filtering result of iPhone Positioning Data Set after 2 iteration

iPhone selects positioning data among cellular base station locations, GPS, and crowd sourced WIFI positioning system. We can conclude that two iterations are sufficient in order to include 75% of positioning data. The rate 75% was chosen to exclude relatively incorrect positioning data from cellular network and crowd sourced WIFI positioning. Once we are certain that we use only GPS data which has 87.7% of accuracy, we can make three iterations in order to filter out 12.3% of incorrect data. Table 1 also contains figure numbers which is corresponding to iteration step. The inclusion of erroneous position can be noticed with more than three iteration. The collector of iPhone data set, actually an author of this paper, verifies figures and thus concludes that every positioning data of error was removed after second iteration.

7. Conclusion and Future Research

For the filtering purpose of erroneous positioning data, we used OCSVM and found some adequate results. We conducted basic experiment both in 3GBS and GPS positioning data, and found 3GBS data shows more error in position.

Comparing figure 1 and figure 15, filtering-out of positioning data can be visualized. One of the problems was overfiltering due to the nature of OCSVM. Our solution was hill climbing approach and experienced proper result. We found that 2 to 4 steps of iteration for hill climbing approach will overcome the over-filtering tendency efficiently.

For the next step, we need to bring up simpler version of filtering since OCSVM approach requires very large amount of computation and thus cannot be used on mobile devices with low computational power. Our aim is to develop a lightweight, real-time filtering algorithm that can be ported on usual mobile devices. For this purpose, a filtering algorithm based on the concept of moving average of positioning data sequence is under development.

We think our approach will help to enhance the quality of many applications and help

Figure 20. Filtering result of iPhone Positioning Data Set after 3 iteration

Figure 21. Filtering result of iPhone Positioning Data Set after 4 iteration

Figure 22. Filtering result of iPhone Positioning Data Set after 5 iteration

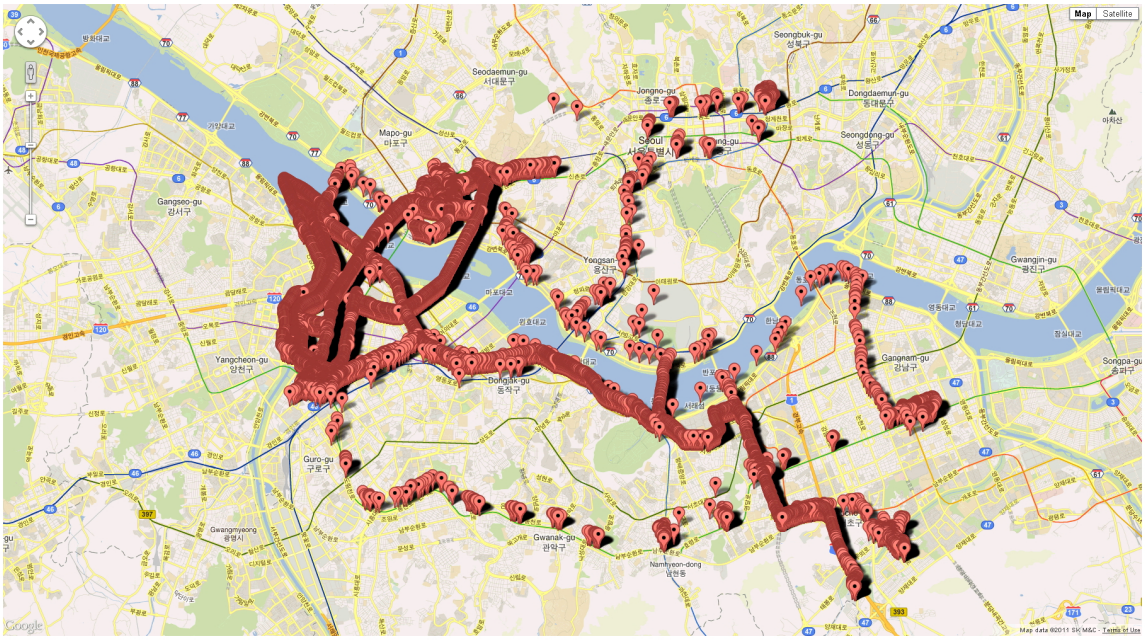

Figure 23. Filtering result of iPhone Positioning Data Set after 6 iteration

many of the researchers as well. We hope our research can be a clue to human mobility modeling. For the construction of accurate human mobility model, our filtering approach can act as a preprocessing stage and filtering out of erroneous data can help the accuracy of human mobility model. For example, we are now doing a research on human mobility modeling based on real positioning data with use of clustering technique. A student's real mobility data is under processing to figure out the daily mobility pattern of the student [16].

References

[1] J. N. Bailenson, M. S. Shum and D. H. Uttal, Road climbing: Principles governing asymmetric route choice on maps, Environmental psychology, 18, 3, pp.251-264 **(1998)**

[2] L. Steg, Car use: Lust and must. Instrumental, symbolic and affective motives for car use, Transportation research part A: Policy and practice, 39, 2-3, pp.147-162 **(2005)**

[3] T. Garling, S. Fujii and O. Boe, Empirical tests of a model of determinants of script-based driving choice, Transportation research part F: Traffic psychology and behaviour, 4, 2, pp.89-102 **(2001)**

[4] L. Steg, C. Vlek and G. Slotegraaf, Instrumental-reasoned and symbolic affective motives for using a motor car, Transportation research part F: Traffic psychology and behaviour, 4, 3, pp.151-169 **(2001)**

[5] B. Verplanken, K. Aarts and A. V. Knippenberg, Habit, information acquisition, and the process of making travel mode choice, European journal of social psychology, 27, 5, pp.539-560 **(1997)**

[6] S. Fujii and T. Garling, Development of script-based travel mode choice after forced change, Transportation research part F: Traffic psychology and behaviour, 6, 2, pp.117-124 **(2003)**

[7] Marta C. Gonzalez and A. Hidalgo and Albert-Laszlo Barabasi, Understanding individual human mobility patterns, Nature, 453 **(2008)**

[8] B. Schlkopf, J. Platt, J. Shawe-Taylor, A. J. Smola and R. C. Williamson. Estimating the support of a high-dimensional distribution. Neural Computation, 13, 7 **(2001)**

[9] Thai Hoang Le and Len Bui, Face Recognition Based on SVM and 2DPCA, International Journal of Signal Processing, Image Processing and Pattern Recognition, 4 **(2011)**

[10] Aborisade, D.O and Adebayo I.G, Cork Stopper Classification Using Feature Selection Method and SVM Based Classifier, International Journal of Signal Processing, Image Processing and Pattern Recognition, 4 **(2011)**

[11] Jung-Chun Liu, Chu-Hsing Lin, Jui-Ling Yu, Wei-Shen Lai and Chia-Han Ho, Anomaly Detection Using LibSVM Training Tools, International Journal of Security and Its Applications, 2 **(2008)**

[12] Garmin GPS Device GPSMAP62s, `https://buy.garmin.com/shop/shop.do?pID=63801`

[13] GPS Real Trajectory, University of Illinois at Chicago, `http://www.cs.uic.edu/~boxu/mp2p/gps_data.html`

[14] iOS 5: Understanding Location Services, `http://support.apple.com/kb/ht4995`

[15] Stuart Jonathan Russell, Peter Norvig, Artificial Intelligence: A Modern Approach

(3rd ed.), Prentice Hall series in artificial intelligence **(2010)**, pp. 122-124, ISBN 0-13-604259-7

[16] H. Kim and H. Y. Song, Daily Life Mobility of A Student: From Position Data to Human Mobility Model through Expectation Maximization Clustering, Communications in Computer and Information Science, 263 **(2011)**

Authors

Woojoong Kim received his B.S. degree in Computer Engineering in 2012 from Hongik University, Seoul, Korea. He is currently a graduate student at Pohang University of Science and Technology, Pohang, Korea. His research interests are in the areas of mobile computing, complex system simulation and machine learning.

Ha Yoon Song received his B.S. degree in Computer Science and Statistics in 1991 and received his M.S. degree in Computer Science in 1993, both from Seoul National University, Seoul, Korea. He received Ph.D. degree in Computer Science from University of California at Los Angeles, USA in 2001. From 2001 he has worked at Department of Computer Engineering, Hongik University, Seoul, Korea and is now associate professor. In his sabbatical year 2009, he worked at Institute of Computer Technology, Vienna University of Technology, Austria as a visiting scholar. His research interests are in the areas of mobile computing, performance analysis, complex system simulation and human mobility modeling.