Technical Report

Department of Computer Science
and Engineering
University of Minnesota
4-192 EECS Building
200 Union Street SE
Minneapolis, MN 55455-0159 USA

TR 02-001

Dictionary Design Algorithms for Vector Map Compression

Shashi Shekhar, Yan Huang, and Judy Djugash

January 05, 2002

Dictionary Design Algorithms for Vector Map Compression

Shashi Shekhar, Yan Huang, Judy Djugash

January 5, 2002

Computer Science Department, University of Minnesota
200 Union Street SE, Minneapolis, MN-55455, USA.
[*shekhar, huangyan, djugash*]*@cs.umn.edu*

Abstract

Vector maps (e.g. road maps) are important in a variety of applications including mobile computing. Due to the large size of vector maps, only a small part of maps (e.g. relevant to current location of the vehicle) can be cached in hand-held or in-vehicle devices used for mobile computing. Compression techniques for vector maps can help cache larger subsets of maps and reduce the communication costs of downloading newer subsets of maps during travel. Dictionary-based compression technique one common means of data compression. This paper explores the problem of designing dictionaries for dictionary based compression techniques for vector maps. We propose a novel clustering-based dictionary design. The proposed approach adapts the dictionary to a given dataset, yielding better approximation. Experimental evaluation shows that when the dictionary size is fixed, the proposed clustering-based technique achieves better accuracy compared with conventional approaches.

Keywords: vector map compression, clustering, dictionary design.

*Supported in part by the Army High Performance Computing Research Center under the auspices of the Department of the Army, Army Research Laboratory Cooperative agreement number DAAH04-95-2-0003/contract number DAAH04-95-C-0008.

1 Introduction

Mobile computing devices, e.g. personal digital assistants (PDA) and in-car navigation units, require access to spatial datasets [5, 10, 11, 12, 13] such as vector maps for location-based services (e.g. where is the nearest gas station?) Vector maps, e.g. road maps, consist of a collection of points (e.g. road intersections), line strings (e.g. center line of road segments connecting intersections), and polygons (e.g. boundaries of parks, cities, lakes etc), in contrast to raster maps, which use image or matrix representation. The enormous size of vector maps and limited storage available in hand-held devices motivate the need for data compression techniques. For example, a typical PDA such as Palm III and Palm V has 2 to 64 mega bytes of storage. The size of a city map (e.g. in Microsoft pocket street format) is usually around .5 to 2 mega bytes. A PDA can store only a few maps in the space remaining after the space occupied by the operating system and other essential software and data.

Compression techniques for vector maps can allow PDAs to carry larger subsets of vector maps or free-up memory for other datasets, e.g. appointments, address book, etc. Compression techniques for vector maps can also reduce the communication cost of downloading new maps to the PDA, possibly over low-bandwidth wireless channels (e.g. beaming, cell phone modems).

The goal of vector map compression is compact representation of map data, with possibly some limited sacrifice of spatial accuracy. Lossy compression schemes are acceptable since cartographers routinely use map simplification to highlight key features in a map by introducing bounded distortions, e.g. errors in the location of spatial objects. Compression schemes for vector maps should allow simple decoding schemes due to the limited computational resources of popular PDAs (e.g. Palm III and Palm V). This constraint makes dictionary-based compression techniques attractive as long as the error of approximation can be controlled.

This paper explores the problem of designing dictionaries for the dictionary based compression of vector map which tries to minimize approximation errors. It proposes the use of clustering techniques[†] (e.g. K-mean clustering [7]) to identify dictionary entries while minimizing errors of approximation for locations of spatial objects in the map. We formally show that this proposed dictionary construction approach often yields a lower error of approximation than the error from conventional fixed dictionary techniques. Experimental results with a road map representing the major US Highways confirm the superiority of the proposed method in yielding lower errors of approximations for a fixed size dictionary.

1.1 Related Work and Our Contributions

Map compression techniques can be divided into two groups, namely raster map compression and vector map compression. Raster map compression techniques manipulate a raster matrix to get a concise representation. Vector map compression includes techniques such as line simplification, chain codes, and dictionary based compression. Line simplification [6, 14] and chain codes [15] are used with paper maps and other line-drawings containing curves. These techniques approximate curves by a sequence of straight-lines. They use straight lines from a fixed collection, e.g. vertical, horizontal, diagonal. Line-simplification and chain-codes often eliminate original points and add new points as a side-effect. FHM (Fibonacci, Huffman, and Markov) is a chain codes based [8, 15] algorithm designed for signature compression. Given a

[†]For an excellent summary of clustering algorithms please refer to [7]

signature, the FHM algorithm uses a dictionary of line segments with fixed slopes and lengths. It replaces each line segment or group of consecutive line segments in the signature by searching the dictionary for the best fitting one.

However, digital vector-maps usually do not contain curves; rather, they consist of points, line-strings and polygons. Adding or removing points may not be allowed in compression since the map accuracy assessment [3] are based on comparison of equivalent points. Despite this constraint, the dictionary approach used in the FHM [9] algorithm can still be used. Dictionary based algorithms construct a dictionary to match dictionary entries to line segments. When a dataset is decompressed, the dictionary is searched to find the data associated with the dictionary indices. The FHM dictionary, as a static dictionary, does not consider the data distribution (slopes and offsets of segments in the curves) but rather uses a collection of line segments organized by a set of squares. If the distribution of line segments in a dataset does not match well with the dictionary entries of FHM, the errors of approximation can be large. In such a scenario, an alternative method which could incorporate the data distribution in the dictionary design would achieve better spatial accuracy than the FHM method.

In this paper, we propose a clustering-based map compression method which adapts a dictionary to a given dataset. This method yields better approximations of the original map data leading to better accuracy. We provide an experiment design and evaluation on a real dataset. Experimental results show that for a fixed size dictionary, the clustering-based dictionary construction compression method achieved better accuracy than the use of a static dictionary such as that used by the FHM algorithm.

1.2 Outline and Scope

In section 2, we discuss issues related to vector map compression. Different dictionary building schemes are described in section 3 followed by the experiment evaluation and comparison in section 4. We conclude in section 5.

The concepts of chain coding and line simplification are beyond the scope of this paper, though if used might lead to better minimization of errors. Also, the scope of this paper is limited to data comprised of line segments. Datasets which include curves are not addressed here.

2 Vector Map Compression Issues and Problem Definition

Typically a vector map is composed of objects represented by points, lines, polygons, and graphs. There are texts associated with each object. In the map, we can observe the metric properties of the objects in the map such as distance as well as topological properties [16] such as crossing, closed, adjacent, and disjoint objects. A map also has a scale and usually supports operations such as zoom in or zoom out.

When a vector map is compressed, the topological properties need to be preserved. Roads which intersect should still appear to intersect after compression and de-compression of the map. When we compress the metric part of a map, the topological part should be preserved as much as possible. Figure 1 shows a framework for obeying this constraints. Vector map data is separated into topological data (e.g. nodes, edges) and geometric data (e.g. sequence of shape points for an edge). Lossy compression techniques may be used for geometric data but not for

topological data. We focus on geometric data from here onwards, as the size of geometric data is usually much larger than the size of topological data.

Figure 1: Map Compression Framework

Compression techniques for geometric data need to observe constraints as well. National Map Accuracy Standards require that maps produced by all Federal agencies comply with published standards such as horizontal and vertical accuracy. For example, the horizontal accuracy standard requires that the positions of 90 percent of all points tested must be accurate within 1/30th of an inch on a map at greater than 1:20,000 scale, 1/50th of an inch on a map at 1:20,000 scale or smaller. Maps produced through compression and de-compression also need to maintain a level of user tolerant accuracy. These accuracy levels differ according to different application domains. Users of roads maps for traveling may have fewer accuracy requirements than users of maps for road construction.

We define the problem of designing a dictionary for vector map compression as follows: Given a vector map and a specific dictionary size, we need to find a dictionary for the dictionary-based encoding scheme. The objective is to minimize the error of approximation for the given data. In achieving this objective, we need to keep in mind the constraints, namely that the topology of the map need to be preserved. For example, road intersections in a road map needs to be maintained. Similarly, if we have a map consisting of all the states in the United States represented as polygons, then these polygons need to remain closed.

3 Dictionary Design Techniques

3.1 Framework

The overall framework for a dictionary based compression technique is presented in figure 2. Given a line string or polygon in an uncompressed map form, we first convert the coordinates in the uncompressed map into a base point followed by a sequence of differential vectors. The differential vectors may be determined by the vector differences between the current point and the previous point or the first point [†]. Now we have a set of O-objects in the form of a BASE point followed by a series of differential vectors. The series of differential vectors produced can then be encoded to produce the compressed data by passing through a two-step process. First a dictionary of a given size needs to be constructed for the produced set of vectors, as shown

Figure 2: Dictionary Based Compression Technique

in Figure 2. In the next step, the dictionary is used to encode the original dataset to produce the encoded, compressed data representation.

To illustrate, consider the road segment given in figure 3. The data set in uncompressed form can be represented as a series of coordinates of the form $(5,5),(6.3,6),(7.6,5),(9.1,8.5)$. Converting this to a base point followed by a sequence of vector representations using the formula $\Delta X = X_i - X_{i-1}$ and $\Delta Y = Y_i - Y_{i-1}$, we get $((5,5),(1.3,1),(1.3,-1),(1.5,3.5))$. Now leaving the base point as it is, the differential vectors $\vec{a}$, $\vec{b}$ and $\vec{c}$ are encoded based on the closest entries from the dictionary. This representation often compounds errors of approximation. For example, errors of approximation for the last point are higher than those for earlier points. A different representation uses differential vectors $(\Delta X_i = X_i - X_0, \Delta Y_i = Y_i - Y_0)$, where (X_0, Y_0) is the first point. The road segment in figure 3 can be represented as $((5,5),(1.3,1),(2.6,0),(4.1,3.5))$. This representation allows direct control of errors of approximation during dictionary construction since the encoding of each differential vector is independent of the encoding of the other differential vectors. This compressed representation along with the chosen dictionary can be used later to decode the data for the purpose of analysis and error computation. In the following subsections, we present two approaches for dictionary construction.

Figure 3: Example of a Road Segment

3.2 Dictionary Based on FHM Curve Compression

The FHM (Fibonacci, Huffman, and Markov) method based on multiring chain coding [9] is designed for compressing signatures. Squares of specific sizes are built around the current anchor point based on Fibonacci numbers. Thus a set of nested squares of sizes 1X1, 2X2, 3X3, 5X5, 8X8 and 13X13 is produced. For the purposes of our dataset, we have used a modified version of the FHM algorithm. We use the Fibonacci series to determine the squares and the dictionary entries. The pseudocode for this version is as given below. First the dictionary is built using selected points on squares S1,S2,S3,S5,S8 and S13, giving us 256 points. Each point in the given dataset is then encoded as one of the dictionary points based on the nearest dictionary value that can represent this point.

Pseudocode for FHM Compression:

- 1) Pick a square size s , for the grid based on dataset
- 2) **for** square size s build static dictionary **do**
- 3) **for** square S1 **do**
- 4) Pick 8 points with x or y distance from grid center as s
- 5) **end;**
- 6) **for** square S2 **do**
- 7) Pick 16 points with x or y distance from grid center as $2*s$
- 8) **end;**
- 9) **for** square S3 **do**
- 10) Pick 24 points with x or y distance from grid center as $3*s$
- 11) **end;**
- 12) **for** square S5 **do**
- 13) Pick 40 points with x or y distance from grid center as $5*s$
- 14) **end;**
- 15) **for** square S8 **do**
- 16) Pick 64 points with x or y distance from grid center as $8*s$
- 17) **end;**
- 18) **for** square S13 **do**
- 19) Pick 104 points with x or y distance from grid center as $13*s$
- 20) **end;**
- 21) **end;**
- 22) **for** Each road segment **do**
- 23) Set the base point of the road segment as the current anchor point
- 24) Pick point P_i on the road segment


```

25)  If  $P_i$  is the first point outside square  $S_n$ 
26)      Get the closest point  $M_i$  in the dictionary which is closest to  $P_i$ 
27)      Encode  $P_i$  to  $M_i$ 
28)      Pick next point  $P_{i+1}$ 
29)      Goto (24)
30)  Else
31)      Pick Point  $P_{i+1}$ 
32)      Goto (3)
33)  end;
34) end;

```

For example, consider the road segment in figure 3. We provide the sequence of vectors created using $(X_i - X_{i-1}, Y_i - Y_{i-1})$ as inputs to the FHM-based compression. Figure 4 shows the result of this compression. The square size is picked to be 1 square unit for this example. The grid points for the first three squares that were generated based on the Fibonacci series are shown here. The differential vector $\vec{a}$ gets mapped onto the dictionary point 1, vector $\vec{b}$ gets mapped onto point 7, and vector $\vec{c}$ onto point 27 in the dictionary respectively.

Figure 4: Example of Encoding by the FHM Algorithm

3.3 Clustering-Based Compression (CBC) Methods

The clustering-based compression (CBC) method uses a clustering algorithm to generate a dictionary of the given size. We have used the K-mean clustering [7] algorithm for dictionary design. K-mean clustering takes as input a fixed number (K) and generates that many clusters for the given dataset as output. The K-mean cluster centroids obtained becomes our dictionary. Based on this dictionary, we encode the vector dataset that we obtained earlier. Since each vector would now be assigned to a particular cluster, that vector would now be represented in terms of a reference to that cluster's centroid entry in the dictionary.

The encoding of each differential vector is the index of the spatially closest dictionary entry. For future decoding purposes, the dictionary is sent along with the encoded data. The pseudocode for this method is as shown below.

CBC Algorithm:

- 1) **for** Each road segment **do**
- 2) Separate base point and sequence of delta values

```

3) end;
4) Do
5)   K-mean clustering on sequence of delta values (using clementine)
6)   for Each original road segment do
7)     Encode road segment using K-mean model built
8)   end;
9) end;

```

The K-mean clustering algorithm provides a solution to the dictionary design problem stated in Section 2. It can be shown that the objective function of K-mean clustering (error of fit between differential vectors and centroids selected) is the same as the objective of the problem of designing a dictionary for differential vector map compression with minimal spatial error. In figure 5, a line segment is represented by its starting and ending points. The delta values are calculated as shown. After we cluster all the delta values ($\Delta x_j^e, \Delta y_j^e$) they error of K-mean is calculated by

$$\sum_j (x_j^e - x_{c_j}^e)^2 + (y_j^e - y_{c_j}^e)^2$$

and the error in real space is calculated by

$$\sum_j (x_j^e - (x_j^s + x_{c_j}^e))^2 + (y_j^e - (y_j^s + y_{c_j}^e))^2$$

assuming that the difference vectors are relative to the first point in the linestring. Since

$$\Delta x_j^e = x_j^e - x_j^s, \Delta y_j^e = y_j^e - y_j^s$$

a comparison of the first two equations reveals that the error resulting from the K-mean clustering approach equals the error in the real dataset.

Figure 5: K-mean Algorithm Error Illustration

Note that K-mean clustering algorithms are often greedy and may not provide optimal solutions, i.e. optimal dictionary minimizing spatial errors of approximation. However, K-mean clustering can provide better dictionaries than those used in FHM. The proof of this lemma using differential vectors built relative to previous points rather than to the first point is part of our future work. However, we believe that the relative impact of this on FHM and K-mean would be the same.

Lemma:

When the dictionary size is fixed, CBC using the starting point of a road as the base point always achieves equal or better accuracy than the modified FHM algorithm.

Proof:

K-mean clustering starts from k random points as the k centroids. It clusters the dataset by

assigning each point to the nearest centroid, recalculates the means of each cluster, and takes the means as the new centroids. If the total square error is decreasing, it iterates until the total square error is non-decreasing. When we compress and de-compress the data by CBC using the starting point of a road as the base point, each point (x_j^e, y_j^e) is represented $(x_j^s + x_{c_j}, y_j^s + y_{c_j})$ where (x_j^s, y_j^s) is the base point and (x_{c_j}, y_{c_j}) is the centroid of the cluster which contains $(x_j^e - x_j^s, y_j^e - y_j^s)$. As we just showed, the error resulting from K-mean clustering is the same as the error produced by the compression using the starting point of a road as the base point. By choosing the entries in the FHM dictionary as the initial K-centroids, and iterating through the K-mean clustering algorithm to achieve smaller total square error, we can guarantee that the CBC, which uses the starting point of a road as the base point, achieves better accuracy.

4 Experiment Design and Results

The dataset included 47,014 road segments from the National Atlas of the United States of America, which provides geographic data in shapefile format. It contains the major roads and highways from the United States Atlas [4]. The shapefiles were converted to Arc/Info [1] coverage using the Arc command "SHAPEARC". Spatial attributes (coordinates) of the coverages were then extracted in a text file using the Arc command "UNGENERATE". The dictionary-based map compression method was applied on this extracted coordinates dataset by first obtaining the delta values for each of these road segments. For this particular dataset, we obtained 40,9964 delta values. The distribution of these delta values are shown in figure 6 (b). K-mean clustering was then done on these delta values using Clementine [2] with the value of K set to 256 so as to build a dictionary of 256-mean values. The original dataset was then encoded using the dictionary built.

Figure 1 presents the experiment design. A given vector map is first decomposed to geometry part and topology part. The geometry data is then converted to delta vectors. There are two parameters to decide how to compute the delta vectors: the co-ordinate system (absolute or relative) and the delta definition (related to the first node of the road segment or related to the previous node of the road segment). FHM or CBC algorithms could be used to design the dictionary using the delta values. Then the dictionary built is used to compress the original dataset. The final step is to evaluate the compression scheme by computing error and visualizing the de-compressed dataset compared with the original dataset.

4.1 Comparison of Dictionary Construction Methods

We evaluate the impact of the co-ordinate system choice and the impact of the delta value choice on both FHM and CBC algorithms and then focus on the impact of delta definition choice on CBC and the impact of co-ordinate system choice on CBC. We give explanations based on the delta distributions and dictionaries constructed.

Figure 7 and Figure 8 show the delta distributions for the combination of the co-ordinate system choice and the delta definition choice with the dictionary entry distributions below them. Whenever possible we will use these two figures to explain the experimental results.

[†]Another interesting way to derive differential vectors is via using relative coordinate system with x-axis defined by the first and last points in the line string. We will explore this in future work.

Figure 6: Dictionary Built Based on FHM Algorithm

The major difference between our technique and the FHM algorithm is that the former uses the K-mean clustering technique to produce the dictionary values, while the latter uses a predefined dictionary. Figures 6 (a) and (c) show the dictionaries built for the above described dataset based on the FHM algorithm and CBC approach respectively. We note that the dictionary designed by CBC closely mimics the distribution of the differential vectors. We also note that the distribution of differential vectors in our dataset is quite different from that assumed by the dictionary used by FHM.

The dataset was encoded using these dictionaries and the root mean square error values were computed for the two approaches. Figure ?? provides a comparison of these error values. We can observe that for this particular dataset, the clustering based dictionary approach works better. The error of approximation achieved by the CBC scheme is much lower than the errors produced by different instances of FHM dictionaries. Hence we could expect that the minimum error would not go below the median of the delta values of the dataset. The experiments that we conducted also show the same result, as can be seen from figure ?. The encoded data that was decoded using both dictionaries was also converted into shapefiles, again using ArcInfo, and visualized using ESRI's ArcExplorer 3.1. Figures 11 (a) and (b) show the visualization of the two approaches superimposed on the original dataset. The fact that the error produced by the FHM-based approach is much larger than that produced by the CBC approach can be explicitly seen from these figures. For example, in the left side of figure 11 (a), we can spot that the deviations between the original road segments and the superimposed ones are much larger as compared to those in 11 (b).

4.1.1 The Impact of Co-ordinate System Choice on FHM and CBC

Figure 9 gives the root mean square error of FHM and CBC with the delta defined based on the previous point in both co-ordinate systems. Because the square sizes affect the FHM dictionary design we evaluate the error of FHM over a wide range of square sizes to ensure the soundness of the experiment. CBC does not use square sizes and the error remains constant as the square sizes increase. We can see that in both co-ordinate systems CBC has a smaller error than that of FHM.

4.1.2 The Impact of Delta Choice on FHM and CBC

How to explain the change of x-axis of the co-ordinate system? How to explain the experiments are not exaustive?

4.1.3 The Impact of Delta Choice on CBC

Table 1 shows the errors of CBC with different delta definitions in an absolute co-ordinate system. K-mean error is different from the root square error of CBC with the delta definition based on the previous point because the errors accumulate when decoding. The CBC achieves better accuracy with the delta definition based on the previous point inspite of the error accumulation in decoding. Here the average number of shape points per road segment is 8. With the number of shape points per road segment increasing, error accumulation might increase for CBC with delta definition based on previous point.

Delta Definition	Coordinate System	Root Mean Square Error	K-mean Error
$\Delta_{i,i-1}$	A	3074	244
$\Delta_{i,0}$	A	6284	6284

Table 1: Root Mean Square Error Values for the Encoding Schemes

4.1.4 The Impact of Co-ordinate System Choice on CBC

Table 2 shows the errors of CBC in different co-ordinate systems and with delta definition based on previous point. The relative coordinate system leads to lower error.

Delta Definition	Coordinate System	Root Mean Square Error
$\Delta_{i,i-1}$	A	3074
$\Delta_{i,i-1}$	R	2092

Table 2: Root Mean Square Error Values for Coordinate system change

5 Conclusion and Future Work

In this paper we have explored the problem of designing dictionaries for dictionary-based vector map compression keeping in mind that the objective is to minimize the error. We have proposed the use of clustering algorithms to produce an adaptive dictionary based on a given dataset. We have also formally shown through a lemma that this proposed dictionary construction approach always yields lower errors of approximation than the conventional fixed dictionary techniques used by the FHM algorithm.

In our future work we would like to experiment on different differential vector schemes. The general purpose clustering algorithm aims at minimizing the total square error. When a user tolerant accuracy level for each road segment is specified, the clustering algorithm needs to be modified to meet this requirement.

Acknowledgments

We thank Raju Vatsavai, Xiaobin Ma, Hui Xiong, and Pusheng Zhang of the University of Minnesota for their valuable feedback on earlier versions of this paper.

References

- [1] Arc/Info 8. <http://www.esri.com/software/arcview/index.html>.
- [2] Clementine. <http://www.spss.com/clementine/>.
- [3] Map Accuracy Standard. <http://mac.usgs.gov/mac/isb/pubs/factsheets/fs17199.html>.
- [4] National Atlas of the United States of America . <http://www.nationalatlas.gov/>.
- [5] S. Chawla, S. Shekhar, and U. Ozesmi. Extending Data Mining for Spatial Applications: A Case Study in Predicting Nest Locations. In *Proc. Int. Conf. on 2000 ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery (DMKD 2000)*, 2000.
- [6] D.H. Douglas and T.K. Peucker. An algorithm for the reduction of the number of points required to represent a digitized line or its character. *The Canadian Cartographer*, 10, 1973.
- [7] A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data*. Prentice Hall, 1988.
- [8] C. C. Lu and J. G. Dunham. Hightly Efficient Coding Schemes for Contour Lines Based on Chain Code Representations. *IEEE Transactions on Communications*, 39(10):1511–1514, Mar. 1991.
- [9] D. Salomon. *Data Compression: the Complete Reference*. Springer-Verlag, 2nd edition, 2000.
- [10] S. Shekhar and S. Chawla. Spatial Databases: Issues, Implementation and Trends. *Prentice Hall (under contract)*, 2001.
- [11] S. Shekhar and Y. Huang. Co-location Rules Mining: A Summary of Results. In *Proc. Spatio-temporal Symposium on Databases*, 2001.
- [12] S. Shekhar, C.T. Lu, and P.S. Zhang. Detecting Graph-based Spatial Outliers: Algorithms and Applications. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2001.
- [13] S. Shekhar, T.A. Yang, and P. Hancock. An Intelligent Vehicle Highway Information Management System. *Intl Jr. on Microcomputers in Civil Engineering (Publisher: Blackwell Publishers)*, 8(3), 1993.
- [14] K. Thappa. Automatic line generalization using zero-crossings. *Photogrammetric Engineering and Remote Sensing*, 54(4), Apr. 1988.
- [15] P. W. Wong and J. Koplowitz. Chain Codes and Their Linear Reconstruction Filters. *IEEE Transactions on Information Theory*, 38(2):268–280, Mar. 1992.
- [16] M.F. Worboys. *GIS: A Computing Perspective*. Taylor and Francis, 1995.

Figure 7: Distribution of Δ Vectors for Benchmark Vector Map and Dictionaries Constructed for CBC vs. FHM of Absolute Coordinate system a) Delta Distribution for $A-\Delta_{i,i-1}$ b) Delta Distribution for $A-\Delta_{i,0}$ c) Dictionary Built Based on K-Mean Clustering Algorithm for $A-\Delta_{i,i-1}$ d) Dictionary Built Based on K-Mean Clustering Algorithm for $A-\Delta_{i,0}$

Figure 8: Distribution of Δ Vectors for Benchmark Vector Map and Dictionaries Constructed for CBC vs. FHM of Relative Coordinate System a) Delta Distribution for $R-\Delta_{i,i-1}$ b) Delta Distribution for $R-\Delta_{i,0}$ c) Dictionary Built Based on K-Mean Clustering Algorithm for $R-\Delta_{i,i-1}$ d) Dictionary Built Based on K-Mean Clustering Algorithm for $R-\Delta_{i,0}$

Figure 9: Impact of coordinate system choice a) Root Mean Square Error for FHM vs. CBC using $A-\Delta_{i,i-1}$ b) Root Mean Square Error for FHM vs. CBC using $R-\Delta_{i,i-1}$

Figure 10: Impact of delta choice a) Root Mean Square Error for FHM vs. CBC using $R-\Delta_{i,i-1}$ b) Root Mean Square Error for FHM vs. CBC using $R-\Delta_{i,0}$

Figure 11: a) Map produced by using FHM Algorithm, superimposed on Original Map b) Map produced by using K-Mean Clustering Algorithm, superimposed on Original Map