

Location and Transportation Services for Yangon Region

MyatThiri Khine^{#1} and MyintMyint Sein^{#2}

[#]Research and Development Department (GIS Lab), University of Computer Studies, Yangon, Myanmar

¹myatthirikhine@ucsy.edu.mm

²myintucsy@gmail.com

Abstract -Location-based services applications are widely used in many developed countries. But it still poor in developing countries as Myanmar. One of the requirements of the cities of Myanmar is to fulfil the effective transportation services and location-based services not only for the local people but also for tourists who are not familiar with transportation system and places in Myanmar. Transportation system in Myanmar is very complex and most of the Myanmar people use the public transportation bus lines. In this proposed system, the desired nearest places in Yangon region can be searched and it also guides where the bus stop is and which buses need to take to get there with public buses. This application can be used on the mobile devices. To quickly retrieve the desired information, a new index structure that combines the inverted is constructed. In this index structure, Hilbert space filling curve and B-tree is used.

Keywords –Location-based service, Transportation services, Hilbert curve, B-tree, Index structure

I. INTRODUCTION

Although the effective Location-based services have been developed in the developed countries, it still lacks to develop the effective and efficient one in the developing countries as Myanmar. Location-based services application for Myanmar has been developed but most of the applications mainly depend on the web services. So, it still needs to develop it on the mobile devices to quickly search the desired location anywhere and anytime.

Most of the citizens in Yangon area make use of public transportation type such as bus, taxi, train, plains and so on. Most of the people in Yangon mainly use the public transportation bus lines. There are over 250 bus lines running around Yangon and over 2.5 million commuters a day. There are nearly 2000 bus stops in Yangon downtown area. Public transportation service is not very easy to travel not only in Yangon, but also for the whole country. So, good transportation services are necessary to move people quickly, comfortably and affordably to their destinations.

Many index structures have been proposed in recent years to quickly retrieve the information. R-tree is mainly used and that combine with inverted file, namely the families of IR-tree [12, 11, 6, 7, 9, 8, 10].

R-tree is used for spatial (latitude/longitude) index and inverted file is used for textual index. As the data objects in the R-tree can be overlapping and covering each other, the search process in the R-tree might suffer from unnecessary node visits and higher IO cost [13]. Hybrid index structure that combines the K-d tree and the inverted file for spatial keyword search with the minimum IO costs and CPU costs has been proposed [1][2][3]. In K-d tree, the data points are scattered all over the tree.

This paper presents the mobile application which can search the desired nearest places and also shows the transit buses list to get the desired location. A new index structure is constructed to effectively and efficiently retrieve the information. It can also reduce the searching time. In this index structure, Hilbert space filling curve and B-tree is used.

The remainder of the paper is organized as follows. Section II reviews the related work. Hilbert Curve is explained in section III. Section IV describes B-tree. In section V, the proposed system is described. The experimental results is showed in section VI. Section VII concludes the paper with directions for future work.

II. RELATED WORKS

There are many index structure has been developed. Most index structures [10, 11, 6, 9, 8] use R-tree and its variants as spatial index and inverted file for text index. They all combine both indices depending on the combination schemes [5]. T.Wang, G. Li, J. Feng [14] proposed a new index structure, spatial keyword R-tree, called SKR-Tree which extended from the R-tree with an R-tree node storing both spatial and keyword information. Hariharan et al. R. Göbel, A. Henrich, R. Niemann, and D. Blank [9] presented the KR*-tree. This paper proposed a framework for GIR systems and focus on indexing strategies. I. D. Felipe, V. Hristidis, and N. Rishe [8] uses R*-tree for spatial index and inverted file for text index. The IR tree [6] creates each nodes of the R-tree with a summary of the text content of the objects in the corresponding subtree. X. Cao, L. Chen, G. Cong, C. S. Jensen, Q. Qu, A. Skovsgaard, D. Wu, and M. L. Yiu [4] proposed S2I index structure based on R-tree and inverted file.

The objects in [4] are stored differently according to the document frequency and infrequency of the term. Cary et al, [11] proposed SKI that combines and R-tree with an inverted index by the inclusion of spatial references in posting lists. The posting list of term contains all its term bitmaps rather than documents

X.Cao, G.Cong, Christian S. Jensen, Jun.J. Ng, BengC.Ooi, N.T. Phan, D. Wu [5] proposes a Web Object Retrieval System (SWORS) that is capable of efficiently retrieving spatial web objects that satisfy spatial keyword queries. This system use IR tree and inverted file for index. It supports two types of queries that are location aware top-k text retrieval (Lkt) query and spatial keyword group (SKG) query.

III. HILBERT CURVE

The Hilbert Curve is space-filling curve which visits every point within a two dimensional space. The basic curve has the shape of an upside down "U". A square is initially divided into 4 quadrants and a first-order curve is drawn through their centre points. The quadrants are ordered such that any two which are adjacent in the ordering share a common edge. The top vertices are replaced by the previous order, and the bottom vertices suffer a rotation. The bottom left vertex is rotated 90 degrees clockwise, and the bottom right rotates 90 degrees counterclockwise. Fig. 1 shows the Hilbert curve orders one, two and three. In this figure, the curve starts on the lower left corner and ends on the lower right corner, but this can be changed as long as the curve keeps the "U" shape.[15]

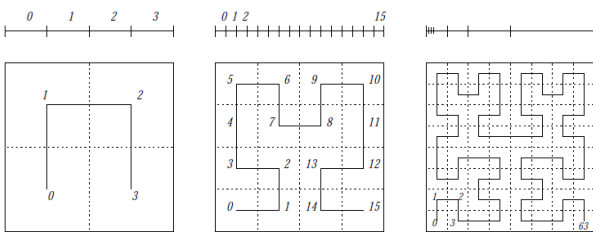


Fig.1 Hilbert Curve orders 1, 2, and 3 respectively.

IV. B-TREE

B-tree is balanced search tree and is similar to red-black trees but they are better at minimizing disk I/O operations. Many database system use B-trees, or variants of B-trees, to store information. B-trees keep values in every node in the tree, and may use the same structure for all nodes. Leaf nodes never have children. Unlike a binary-tree, each node of a b-tree may have a variable number of keys and children. Each key has an associated child that is the root of a sub tree containing all nodes with keys less than or equal

to the key but greater than the preceding key. A node also has an additional rightmost child that is the root for a subtree containing all keys greater than any keys in the node.

A B-tree of order m (the maximum number of children for each node) is a tree which satisfies the following properties:

1. Every node has at most m children.
2. Every node (except root and leaves) has at least $m/2$ children.
3. The root has at least two children if it is not a leaf node.
4. All leaves appear in the same level, and carry information.
5. A non-leaf node with k children contains k-1 keys. [16]

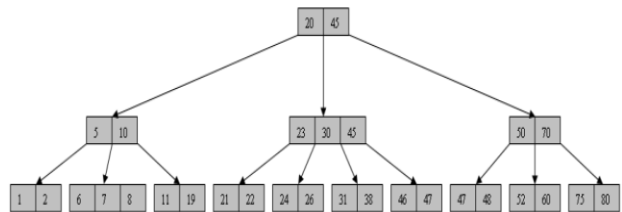


Fig. 2 B-Tree of order 2

V. PROPOSED SYSTEM

The proposed system creates the index structure that combines B-tree with inverted files. Before creating the B-tree, two dimensional coordinate points are converted to the single value by using the Hilbert curve. Then, B-tree that combines the inverted file is constructed according to the value from the Hilbert curve (h-values) and services.

The algorithm to compute the h-values of the two-dimensional Hilbert curve on a $2^n \times 2^n$ grid is:

1. Read in the (n-bit) binary representation of the x and y coordinates.
2. Interleave bits of the two binary numbers into one string, i.e., the same way as for the Peano curve.
3. Divide the string from left to right into 2-bit strings, s_i for $i=1, \dots, N$.
4. Give a decimal value, d_i , for each two bit string according to the following chart
 - a. '00' equals 0
 - b. '01' equals 1
 - c. '10' equals 3
 - d. '11' equals 2

and put into an array in the same order as the strings occurred. (This gives the h-values of the basic Hilbert curve.)

5. For each number i in the array, if $i=0$ then switch every following occurrence of 1 in the array to 3 and every following occurrence of 3 in the array to 1; $i=3$ then

switch every following occurrence of 0 in the array to 2 and every following occurrence of 2 in the array to 0; (This makes up for the rotation and reflection of the curves of order higher than 1.)

- Convert each number in the array to its binary representation (two-bit strings), concatenate all the strings in order from left to right, and calculate the decimal value.[15]

The proposed index structure is shown in Fig. 3. The Example Dataset is shown in Table 1.

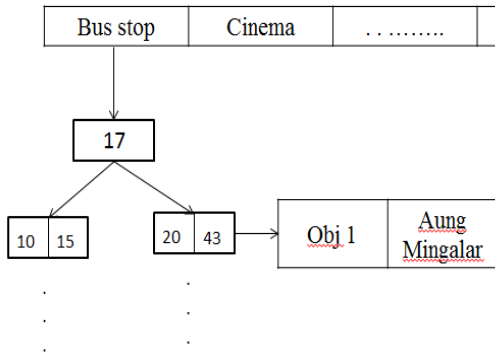


Fig. 3 Proposed Index Structure

TABLE I
EXAMPLE DATASET

id	Latitude	Longitude	Keywords	Services
Obj1	16.796433	96.176803	Aung Mingalar	Bus Stop
Obj 2	16.779908	96.140056	Bahosi	Bus Stop
Obj2	16.800442	96.162225	Bogyoke	Park
Obj3	16.829281	96.155644	Sedona	Hotel
Obj6	16.816497	96.127464	Fuji, Japanese	Restaurant
Obj7	16.810881	96.176419	Shwe Baho	Cinema

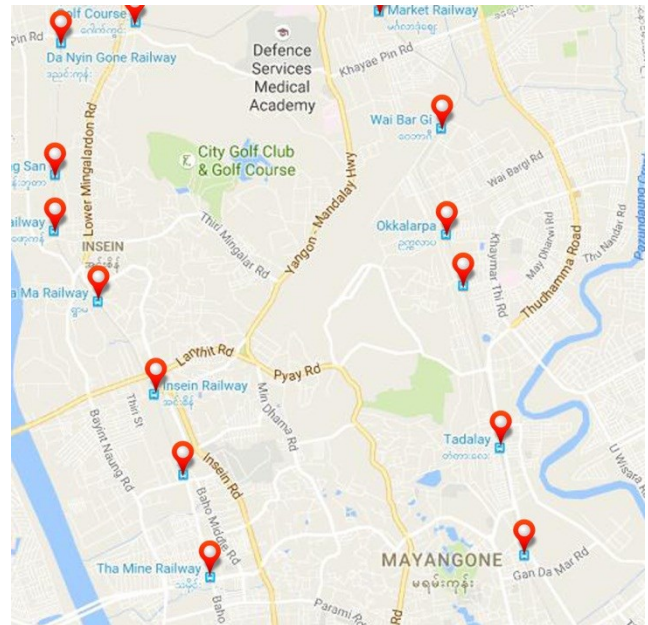


Fig. 4 Bus Stop in Yangon Region

VI. EXPERIMENTAL RESULT

The proposed system searches the desired location based on the input current location and services. Then, it displays the nearest desired location and the list of bus to get there. This system is considered on the mobile devices and is tested on the Yangon Region which has 46 townships. In this system, it is mainly focused on the 23 townships. It provides the user with 72 services and there are 3000 data in the database. Fig. 4 shows the bus stop in Yangon region.

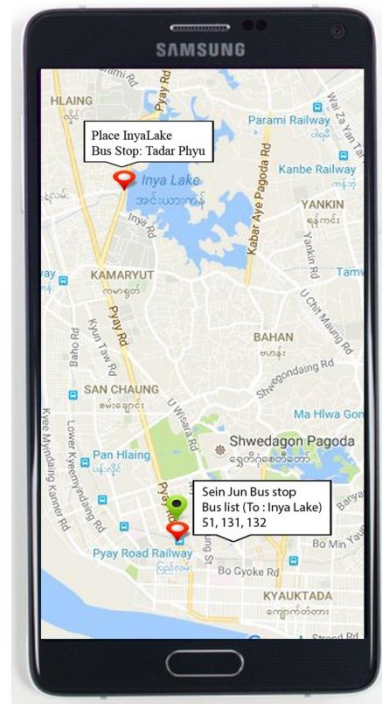


Fig.5 Result of the proposed system

Fig. 5 shows the result of the proposed system and Fig. 6 compares the searching time (second) between using proposed index structure and without using proposed index. Searching time using proposed index structure is 100-times faster than R-tree about 100-times in second.

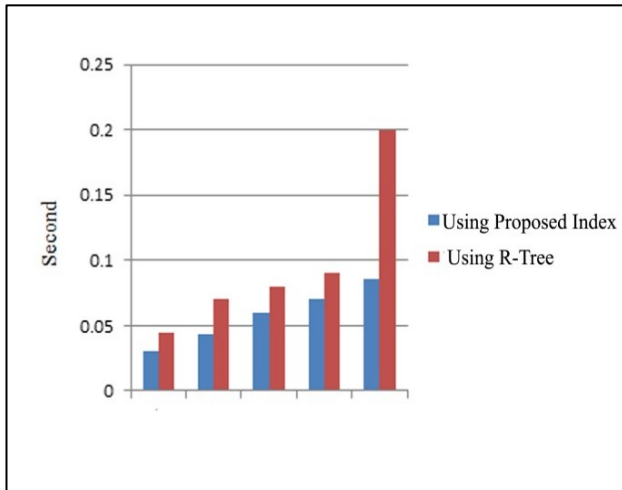


Fig. 6 Searching time compare with proposed index and R-Tree

VII. CONCLUSIONS

In this paper, the proposed system is tested on Yangon region. It searches the desired nearest location and also displays where the nearest bus stop is and which public buses need to take to get the destination. To quickly retrieve the information, the new index structure is constructed using Hilbert curve, B-tree and inverted file. This application is considered on the mobile devices. As a further extension, we will consider the system that will search the desired location with Myanmar Language on the mobile devices.

REFERENCES

[1] S. N. Aung, M. M. Sein, "Hybrid Geo-Textual Index Structure for Spatial Range Keyword Search", *Computer Science & Engineering: An International Journal (CSEIJ)*, Vol. 4, No.5/6, December 2014.

[2] S. N. Aung and M. M. Sein, "K-Nearest Neighbours Approximate Keyword Search for Spatial Database", in *Proceedings of*

9th International Conference on Technological Advances in Electrical, Electronics and Computer Engineering (ICTAECE), Bangkok, Thailand, 7th February 2015, pp. 65-68.

[3] S. N. Aung and M. M. Sein, "Index Structure for Nearest Neighbors Search with Required Keywords on Spatial Database", the *9th International Conference on Genetic and Evolutionary Computing (ICGEC 2015)*, Yangon, Myanmar, 26-28 August 2015.

[4] X. Cao, L. Chen, G. Cong, C. S. Jensen, Q. Qu, A. Skovsgaard, D. Wu, and M. L. Yiu, "Spatial keyword querying", in *ER*, 2012, pages 16-29.

[5] X. Cao, G. Cong, Christian S. Jensen, Jun.J. Ng, BengC.Ooi, N.T. Phan, D. Wu, "SWROS: A System for the Efficient Retrieval of Relevant Spatial Web Objects". Available: <http://www.ntu.edu.sg/home/gaocong/papers/vldb12swors.pdf>

[6] A. Cary, O. Wolfson, and N. Risse, "Efficient and scalable method for processing top-k spatial Boolean queries", in *SSDBM*, 2010, pages 87-95.

[7] G. Cong, C. S. Jensen, and D. Wu, "Efficient retrieval of the top-k most relevant spatial web objects", *PVLDB*, 2009, 2(1):337-348.

[8] I. D. Felipe, V. Hristidis, and N. Risse, "Keyword search on spatial databases", in *ICDE*, 2008, pages 656-665.

[9] R. Göbel, A. Henrich, R. Niemann, and D. Blank, (2009), "A hybrid index structure for geo-textual searches", in *CIKM*, 2009, pages 1625-1628.

[10] R. Hariharan, B. Hore, C. Li, and S. Mehrotra, (2007), "Processing spatial-keyword (sk) queries in geographic information retrieval (gir) systems", in *SSDBM*, 2007, page 16.

[11] Z. Li, K. C. K. Lee, B. Zheng, W.-C. Lee, D. L. Lee, and X. Wang, "I-tree: An efficient index for geographic document search", *IEEE TKDE*, 2011, 23(4):585-599.

[12] J. B. Rocha-Junior, O. Gkorgkas, S. Jonassen, and K. Nørveg, "Efficient processing of top-k spatial keyword queries", in *SSTD*, 2011, pages 205-222.

[13] Y. Theodoridis, T. Sellis, "Optimization Issues in R-tree Construction", Technical Report KDBSLAB-TR-93-08.

[14] T. Wang, G. Li, J. Feng, "Efficient Algorithms for Top-k Keyword Queries on Spatial Databases", *12th IEEE International Conference on Mobile Data Management*, 2011.

[15] Christos Faloutsos, Shari Roseman, "Fractals for Secondary Key Retrieval", In *Proceedings of the Eighth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems ,PODS '89*, 247-252.

[16] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction To Algorithm", Third Edition, pg 484-499