Rendering Process of Digital Terrain Model on Mobile Devices

Siti Aida Mohd Isa saidamisa@gmail.com
ViCube Lab
Department of Computer Graphics and Multimedia
Faculty of Computer Science and Information System
Universiti Teknologi Malaysia
Johor, 81310, Malaysia

Mohd Shafry Mohd Rahim shafry@utm.my
ViCube Lab
Department of Computer Graphics and Multimedia
Faculty of Computer Science and Information System
Universiti Teknologi Malaysia
Johor, 81310, Malaysia

Daut Daman daut@gmail.com
Faculty of Computing and Information Technology
King Abdul Aziz University
Rabigh, Saudi Arabia

Mohd Daud Kasmuni dkasmuni@yahoo.com.uk
Department of Computer System and
Communication, FSKSM
Universiti Teknologi Malaysia
Johor Bahru, 81310, Malaysia

Abstract

Digital Terrain Model has been used in many applications especially in Geographical Information System. However with the recent improvement in mobile devices that can support 3 Dimension (3D) content, rendering 3D based terrain on mobile devices is possible. Although mobile devices have improved its capabilities, rendering 3D terrain is tedious due to the constraint in resources of mobile devices. Furthermore, rendering DTM add more constraint and issues to the mobile devices. This paper focuses on the rendering process of DTM on mobile devices to observe some issues and current constraints occurred. Also to determine the characteristic of terrain properties that will affect the rendering performance. Experiments were performed using five datasets that derived from aerial images. The experimental results are based on speed of rendering and the appearance of the terrain surface. From these results, issues and problems that are highlighted in this paper will be the focus of future research.

Keywords: Mobile Device, Terrain Rendering, Digital Terrain Model.

1. INTRODUCTION

Terrain defined as an area of Earth's surface with distinctive geological features that has been an active area of research since the late 1970s until now. One of the major concepts in representing terrain surface is Digital Terrain Model (DTM). DTM is representations of continuous surface of the ground by a large number of points consist of x, y, z coordinates in an arbitrary coordinate field [1] and are obtainable via satellite, stereoscopic aerial images and contour data from geographical maps before it is saved as a certain file format. DTM has been used as one of the important component in many applications especially in the field of simulation, visualization, games and Geographical Information System (GIS).

The evolution of communication technology and hardware has made researchers in the area of GIS and interactive entertainment field such as games to shift their interest in using mobile devices as a platform. This is due to its mobility which allows user to access and manipulate data at any time and any place. Also, with the recently improved of mobile devices capabilities especially in graphics hardware that can now support 3D based content, the popularity of mobile devices is increased [2, 3]. However, in spite of its capabilities, mobile devices has several limitation like small screen sizes, low-bandwidth, colour resolution, limited processing power, small memory, critical power consumption and its limited application capabilities [2, 3]. Moreover, rendering terrain on mobile device has its own issues because terrain data is naturally large in size.

This paper focuses on the rendering process of DTM on mobile devices. A prototype of terrain rendering on mobile devices is developed and will be tested in a mobile device using five datasets to observe issues and constraints that has been addressed previously. With the experiment, these issues can be clearly understood and method to rectify the problems will be proposed in future research. Discussion of this paper is as follows: Section 2 provides related work concerning terrain rendering. Section 3 discussed the rendering process of DTM on mobile device, followed by Section 4 which provides the information of the experiment for testing purposes. Section 5 shows the result of the experiment and finally, conclusion and future works will be stated in section 6.

2. RELATED WORK

In the last decade, a lot of research and work has been done in the terrain rendering domain ranging from computer to mobile platform. There are two group of family we can identify based from all the terrain rendering method. First category of method is for terrain models that are fit in the memory while another consist of algorithms design to render large terrain data which cannot be completely loaded in memory (out-of-core technique).

In Memory Technique

The management of triangulated irregular network (TINs) are used in most of the approaches and the refinement of mesh in real-time is done by different strategies. Real-time continuous LOD algorithm (CLOD) an early work of Lindstrom, make used of regular-grid representation for terrain rendering [4]. This algorithm conceptually bottom-up mesh reduction defined by right triangles recursively subdivided. The mesh refinement is according to user-specific image quality metric. In parallel to CLOD, Cohen proposes a solution for ray tracing height field that used Delaunay triangulations [5]. Then in 1998, Röttger continue the earlier work of Lindstrom [6] and proposed a geomorphing algorithm to reduce the vertex popping effect [7]. Progressive meshes (PM) are introduce by Hoppe [8] and later extended its application to terrain rendering [9]. Despite the drawback of highly CPU cost, the PM solution can be expended to perform streaming. A year after Lindstrom, Duchaineau publishes Real-Time Optimally Adapting Mesh (ROAM) that proved to be an extremely popular algorithm particularly among game developer [10]. This algorithm used the combination of incremental priority-based approach with binary triangle tree (bintree) structure to optimize the mesh. According to Blow [4], the implementation of ROAM by Dunchaineau is tedious. Some modification is proposed by Blow to improve the ROAM algorithm. Applying a top-down approach and a new error metric using the full three dimensions of source data in performing LOD computations to produced efficient split and merge determinations for high-detail terrain [4].

Rendering 3D graphics on mobile devices is still considered a difficult task even though with the evolution of mobile devices. Research has been done in the area of mobile rendering mostly for 3D games and 3D applications. Previous work on 3D rendering of terrain involves placing pre-computed blocks of terrain together with Perlin noise to represent random terrain on resource limited device [11]. Perlin noise is used to generate a pseudo-random appearance of natural effect. Using this effect is only a subsection of the terrain and it is stored in memory at any one time. This method reduces the amount of memory required for terrain storage.

Joachim and Jean-Eudes [12] proposed multi-resolution representation using strip mask for adaptive rendering of each visible tile. This method saved CPU and memory consumption by transferring the load on 3D graphics device. Method which consists in drawing a planar shadow under each tile is proposed to handle cracks that occurred because of the difference of mask level between two adjacent tiles. The planar shadow is made of two triangles and texture-mapped with the same texture as its corresponding tile. However, even though this technique is fast and simple to implement, it is not a perfect solution and fails in certain cases.

Few years later, Jiang Wen [13] proposed a multi-resolution modelling to represent terrain based on quad-tree. Terrain is divided into regular tiles and represented by hierarchical quad-tree data structure. Then, level-of-detail of each tile is computed and generated dynamically by subdividing based on a set of criteria. The subdivision is according to terrain where fluctuate area is refined and even area is represented by coarse mesh. To eliminate cracks, triangles which are laid on boundary of coarse resolution tile are divided compulsively. This method to handle cracks is still not perfected yet but it is fast and simple to implement. The method proposed has frame rates between 7 to 8 frame per second (fps) by simplifying rendering scene rely on the surface of terrains but the method would not be very effective in mountain area.

Out of Core

In corresponding to our aim to implement terrain rendering in mobile platform, other approaches either propose to perform out-of-core rendering or streaming of the terrain models are reviewed. Out-of-core or external memory address issues related to the hierarchical nature of memory structure of modern computer (fast cache, main memory, hard disk, etc.).

It is important to manage and make the best use of the memory structure when dealing with large data structures that do not fit in the main memory. The out-of-core of the large-scale terrain system presented by Pajarola extends the restricted quadtree triangulation of Lindstrom [14] with another vertex selection algorithm and a more intuitive triangle strip construction method combine with dynamic scene management and progressive meshing [6]. A technique describe by Cignoni named batch dynamic adaptive mesh (BDAM) algorithm for out-of-core management and rendering of large textured terrain [15]. P-BDAM and C-BDAM is an article that extends BDAM which improve in the out-of-core management, data compression, spherical terrains support, a better crack-fixing method, triangle stripping and texture mapping integrated with the geometry LOD management. Regardless of GPU usage in storing BDAMs and implementation of compression, this solution is still unpractical for terrain rendering in mobile platform since they present high CPU cost.

3. TERRAIN RENDERING PROCESS

Developing terrain rendering on mobile devices can be decomposed in two phases. The first phase is to generate suitable input through series of processed in data pre-processing. The second phase is consists of processes that aim to render the terrain surface on mobile device.

Data Pre-processing

Firstly in order to obtain suitable results, pre-processing phase needs to be done using ArcGIS software. Pre-processing phase involved several processes to digitize the aerial images of real world data that are in raster form into vector. These aerial images are in raw format where the images do not contained proper geo-referencing. In fact, the raw data may contain noise. To rectify this issue geo-referencing process needs to be applied which consist of scaling, rotating, translating and de-skewing images to match the desired size and position. Images that have been corrected is processed to generate Digital Terrain Model (DTM) and triangulated into TIN structure to obtain its vertex points. These points each consist of its own coordinate and elevation data in the form of XYZ coordinate where Z point is the height value. Finally, the vertex points are rearranged into a Triangulated Irregular Network (TIN) based file format and will be used as an input for the experiment.

Rendering Process on Mobile Device

In developing terrain rendering on mobile device, terrain surface will be represented as triangulated mesh in 3D space. The triangulated mesh is obtained from the pre-processing phase where the digitized data is produced and arrange into a TIN file. The data file is then loaded line by line into memory after the initialization of the prototype.

Once all the vertex points are loaded, process of calculation for normalization of each point is performed. The data is calculated to find the normal vector of each corresponding point. Afterwards, backface culling is executed using the normalized data to determine the visibility of polygon face as well as the removal of all faces that cannot be seen by the viewer. This process of culling unwanted vertices will reduce the time to render the terrain onto mobile device.

Finally, at the final step of the rendering process, the normalized data that has gone trough the process of backface culling is rendered on the mobile device. Figure 1 illustrates the processes taken to develop terrain rendering on mobile device.

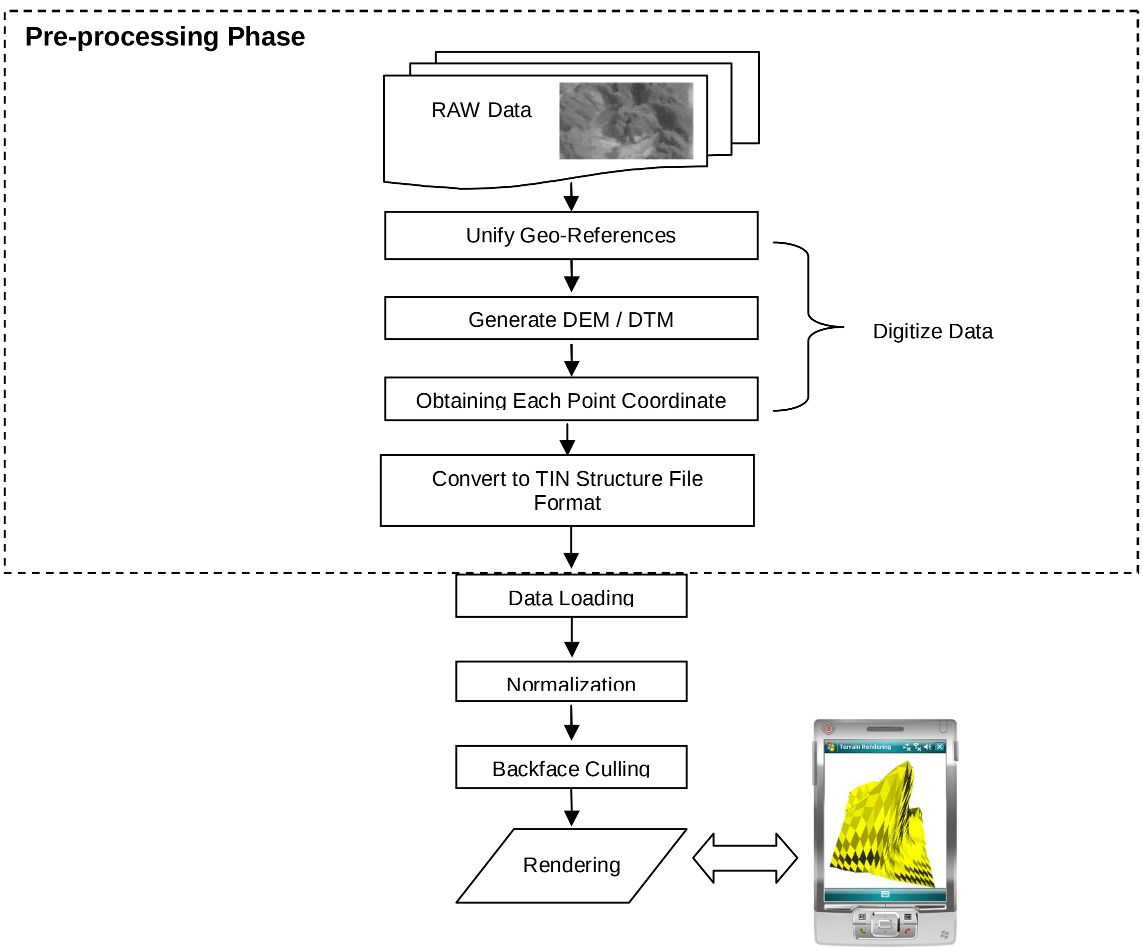


FIGURE 1: Steps of Rendering Process on Mobile Devices.

4. EXPERIMENT

After the development, the prototype will be tested on a mobile device with several terrain dataset.

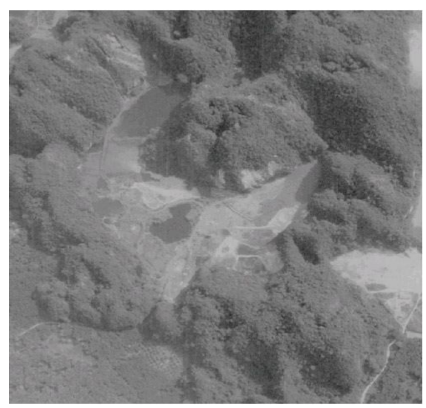
Terrain Dataset

The terrain dataset is obtained from Jabatan Ukur dan Pemetaan Negara (JUPEM) which involved several aerial images of Sungai Kinta region, Perak, Malaysia, near the Banjaran Titiwangsa taken from year 1981. Several pre-processing stages need to be run through by these images before the data can be used as an input.

(a)

(b)

(c)



(d)

(e)

FIGURE 2: Aerial Images of RAW Data Before Pre-processing.

Five datasets are used as an input for the experiment. Detailed properties for these datasets are revealed in Table 1.

Properties	Terrain Data				
	Data A	Data B	Data C	Data D	Data E
Triangles / Faces	836	1150	1302	3528	3528
Vertex	460	624	676	1849	1849
Size on Disk	36Kb	40Kb	40Kb	144Kb	144Kb

TABLE 1: Properties of Each Data in TIN Structure Format.

Testing Platform

In this experiment, the mobile device used as a testing platform is HTC TyTN II while a computer is used for the comparison of the terrain appearance issue. Released in 2007, HTC TyTN II is a Windows Mobile Pocket PC Smartphone designed and market by High Tech Computer Corporation of Taiwan. Table 2 below list most of the important specifications for HTC TyTN II mobile device whilst configurations of the computer are in Table 3
.

Processor	Qualcomm MSM7200, 400MHz
Memory	ROM:250MB RAM:128MB SDRAM
Display	2.8 inch, 240 X 320 QVGA TFT-LCD display with adjustable angle and backlight
Operating System	Windows Mobile 6.1 Professional
Expansion Slot	microSD™ memory card (SD 2.0 compatible)

TABLE 2: Specification of HTC TYTN II Mobile Device.

Processor	AMD Turion (tm) 64 X 2 TL-58, 1.60GHz
Memory	RAM:1.93GB
Resolution	1280 x 800
Operating System	Windows XP

TABLE 3: Specification of the Computer Used for Comparing the Terrain Appearance.

5. EXPERIMENTAL RESULT

This paper discussed the experimental results of rendering terrain on mobile device. The experiments were conducted using five different terrain dataset that were explained in Table 1. These results obtained from running the prototype on the mobile device with specification specified in Table 2. The experimental results and discussion are based on frame per second and the appearance of the terrain surface on the display screen measured by comparing results between on computer and on the device.

Frame per Second

In this paper, 30 second at the beginning of running the prototype is taken from the results of the experiment. Figure 4 shows the experimental results of rendering terrain on HTC TyTN II mobile device.

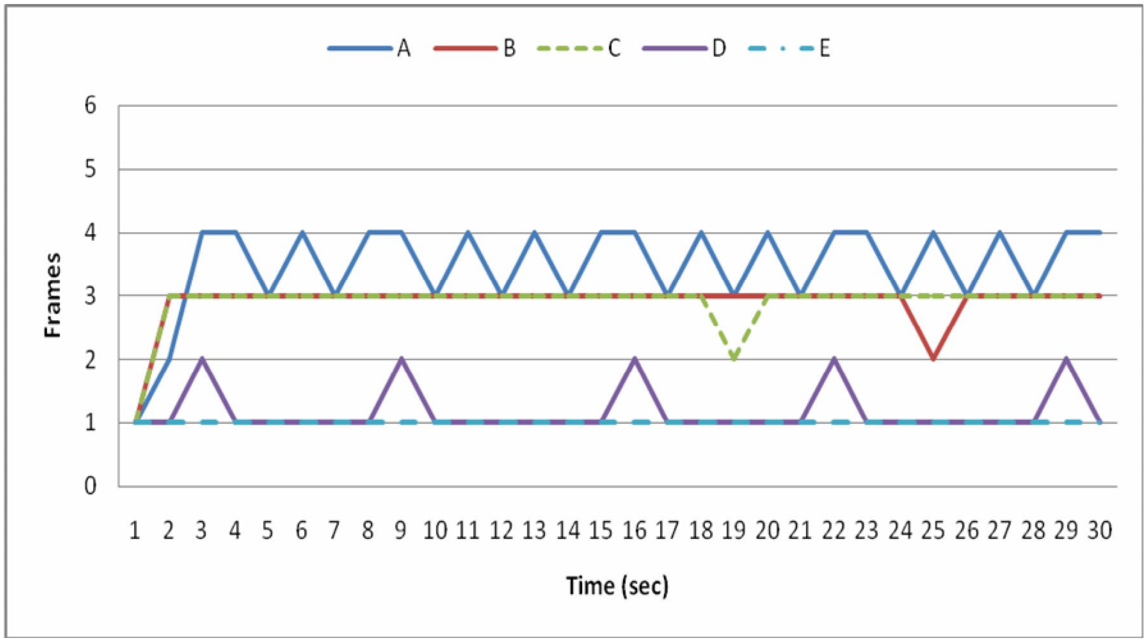


FIGURE 3: Experimental Results for HTC TYTN II Mobile Device.

Rendering process as shown in Figure 1, which includes normalization and backface culling process are to optimize the rendering speed. Normalization is used to find the normal vector of each corresponding point and then backface culling used the normalized data to determine the visibility of polygon face. Backface culling will removed all faces that cannot be seen by the viewer and will render fewer triangles thus will reduce the time to render the terrain onto the mobile device giving the results in Figure 3.

	N	Range	Minimum	Maximum	Mean
A	30	3	1	4	3.47
B	30	2	1	3	2.90
C	30	2	1	3	2.90
D	30	1	1	2	1.17
E	30	0	1	1	1.00
Valid N (listwise)	30				

TABLE 4: Descriptive Statistics.

Based on Table 4, the minimum speed of frame rate for all the dataset is 1 fps. Figure 3 shows that the minimum value of 1 frame is at the beginning of timeline where the prototype in the process of loading the dataset. After the loading, all the dataset have different fps. Data A is between 3-4 fps whilst frame rate within 2-3 fps is the result after running the prototype using Data B and C as an input. Data D has a value of 1-2 fps while E which has the lowest value of 1 fps.

From Figure 3, Data A has a much higher value than other dataset. As explained in Table 1, Data A has 836 triangle and 460 vertices. Additionally, Data A only occupied 36Kb space in disk than other data. Opposite of Data A, Data E has the lowest speed of rendering of 1 fps. Properties of Data E as described in Table 1, the data has 3528 triangle, 1849 vertices and occupied 144Kb of disk space.

Result of Data B and C in Table 4 shows that both data have the same range of 2 and a maximum value of 3 fps. While in Figure 3 shows that the fps is nearly the same pattern with a difference at 19 and 25 second which the frames generated is 1 frame less than the other. Based on the properties of the dataset as shown in Table 1, Data B and C have only a difference of 52 vertices. With such little difference in vertex count, the performance of rendering speed is not affected as shown in Figure 3.

As shown in Figure 3, Data D and E have a different pattern while the range and the maximum shows in Table 4 also have a different value. Although both Data D and E have the same properties (refer Table 1), both have a difference in terms of complexity of the terrain surface. Therefore, the complexity of the terrain is one of the factors that can affect the performance of the rendering speed as shown in Figure 3. From the results observation, Data 4 is the largest in overall which lead to the outcome of the results shown in Figure 3 where the frame rate of Data 4 is lower than the others.

Terrain Appearance

Terrain with realistic and accurately smooth appearance is an important component in some of visualization applications mainly in GIS applications. According to Li (1990), there are three main attribute to be considered; accuracy, efficiency and economy [1]. To determine the appearance of the terrain surface, the experimental results are measured by comparing the appearance of the terrain between results on the mobile device and on computer using the same datasets. The datasets is in the format of TIN structure obtained by rearranging the vertex points derived from the pre-processing stage. Figure 3 illustrate the experimental results for five datasets; Data A, B, C, D and E. The results are obtained where on the left side is from the experiment running on computer whilst on the right is from running on the mobile device.

FIGURE 4: Comparison Based on Appearance for Data A.

FIGURE 5: Comparison Based on Appearance for Data B.

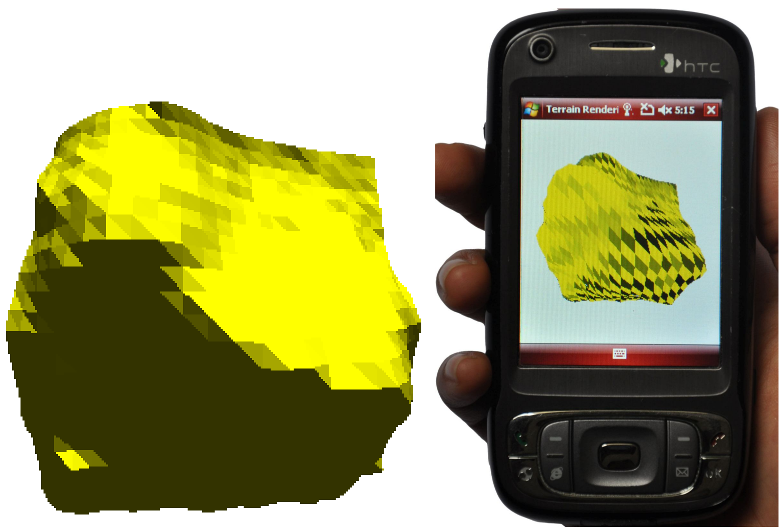


FIGURE 6: Comparison Based on Appearance for Data C.

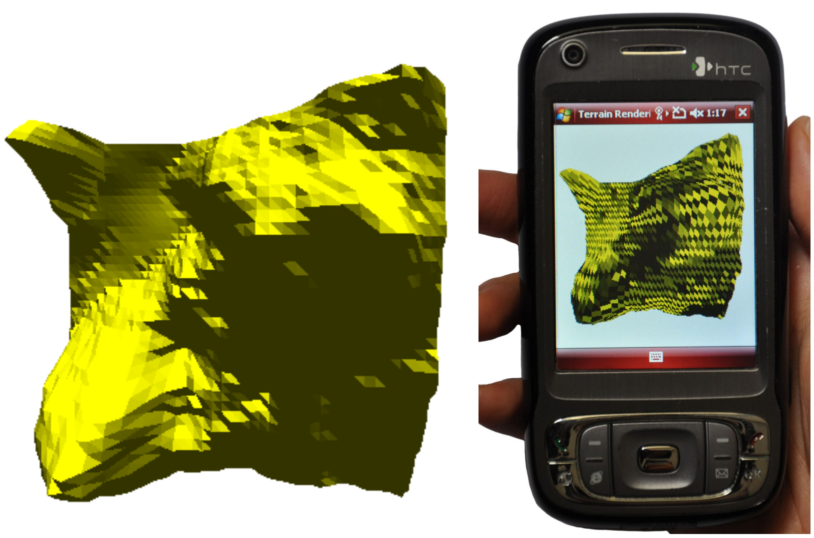


FIGURE 7: Comparison Based on Appearance for Data D.

FIGURE 8: Comparison Based on Appearance for Data E.

In terms of the overall appearance, the results on both computer and mobile device are quite similar. However because of the differences in resolution between mobile device and the computer, the terrain appearance on mobile device looks flat and unrealistic. Based on Table 3, the computer has a resolution of 1280 x 800 whereas 240 x 320 is the resolution for the mobile device.

6. CONCLUSION & FUTURE WORK

In this paper, rendering process on mobile device is described and experiment was performed using HTC TYTN II mobile device with several different datasets. The experimental results shows that the performance of rendering speed even for Data A which its overall properties are much lesser and smaller than the others, has a maximum of 4 fps. This performance result of rendering only 836 triangles is still considered too low even with backface culling is used. Therefore rendering terrain on mobile devices needs to consider many aspects such as the nature of the terrain which is large in size and the complexity of the terrain data itself. Moreover, achieving the acceptable frame rate and accuracy of appearance for rendering terrain depends on how the limitation of mobile devices such as memory capacity and low CPU speed is deal with.

Future work will involve finding an appropriate method or technique that can render more accurate appearance of the terrain surface with an acceptable performance of rendering speed that suitable for limited devices such as mobile device. Towards this goal, level of detail (LOD) for speed or subdivision for accurate appearance should be implemented.

7. REFERENCES

[1] L. Zilin, Z. Qing, and G. Chris "*Digital Terrain Modeling: Principles and Methodology".* CRC Press, 25(1):6-165, 2004

[2] F. Chehimi, P. Coulton, R. Edwards. *"Evolution of 3-D games on mobile phones"*. In Proceedings of the IEEE Fourth International Conference on Mobile Business. Sydney, Australia, 11-13 July 2005

[3] F. Chehimi, P. Coulton, R. Edwards. "*Advances in 3D graphics for Smartphones"*. In: Proceedings of international conference on information and communication technologies: from theory to applications. Damascus, Syria, 24–28, April 2006

[4] J. Blow *"Terrain Rendering at High Levels of Detail"*. In Proceedings of the 2000 Game Developers Conference. 2000

[5] D. Cohen-Or, Y. Levanoni, *"Temporal continuity of levels of detail in Delaunay triangulated terrain"*. In: Proceedings of Visualization '96, pp. 37–42. IEEE Computer Society Press, Los Alamitos, CA. 1996

[6] P. Lindstrom, D. Koller, W. Ribarsky, L F Hodges, N Faust, and G Turner. *"Real-Time, Continuous Level of Detail Rendering of Height Fields"*. In Proceedings of SIGGRAPH 96. pp. 109–118.1996

[7] T. Ulrich *"Rendering massive terrains using chunked level of detail control"*. In SIGGRAPH Course Notes. 2002

[8] H. Hoppe, *"Progressive Meshes"*. In Proceedings of SIGGRAPH 96. pp. 99–108. 1996

[9] H. Hoppe, *"Smooth view-dependent level-of-detail control and its application to terrain rendering".* In Proceedings of Visualization '98, pp. 35–42. 1998

[10] M. Duchaineau, M. Wolinsky, D. E. Sigeti, M. C. Miller, C. Aldrich, and M. B. Mineev-Weinstein. *"ROAMing Terrain: Real-Time Optimally Adapting Meshes"*. In Proceedings of IEEE Visualization '97. pp. 81–88. 1997

[11] D. Marshall, D. Delaney, S. C. McLoone, and T. Ward., *"Representing random terrain on resource limited devices"*. In CGAIDE 2004 Int. Conf. Computer Games: Artificial Intelligence, Design and Education, 2004.

[12] P. Joachim and M. Jean-Eudes, *"Adaptive Streaming and Rendering of Large Terrains using Strip Masks"*. In Proceedings of the 3rd international conference on Computer graphics and interactive techniques. Australasia and South East Asia Dunedin, New Zealand: ACM, 2005

[13] J. Wen, B. Zhu, and F. Wang, *"Real-Time Rendering of Large Terrain on Mobile Device"*, In The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. vol. XXXVII. Part B5. Beijing 2008

[14] R. Pajarola, *"Access to Large Scale Terrain and Image Databases in Geoinformation Systems"*. Swiss Federal Institute of Technology (ETH) Zurich: Ph.D. Thesis. 1998

[15] P. Cignoni, F. Ganovelli, E. Gobbetti, F. Marton, F. Ponchio, and R. Scopigno, 2003b. *"Planet-Sized Batched Dynamic Adaptive Meshes (P-Bdam)"*. In IEEE Visualization 2003.