반도체디스플레이기술학회지 제18권 제1호(2019년 3월)
Journal of the Semiconductor & Display Technology, Vol. 18, No. 1. March 2019.

A Study of the Performance Prediction Models of Mobile Graphics Processing Units

Cheong Ghil Kim[*†]

[*†] Department Of Computer Science, Namseoul University

ABSTRACT

Currently mobile services are on the verge of full commercialization ahead of 5G mobile communication (5G). The first goal could be to preempt the 5G market through realistic media services utilizing VR (Virtual Reality) and AR (Augmented Reality) technologies that users can most easily experience. Basically this movement is based on the advanced development of smart devices and high quality graphics processing computing power of mobile application processors. Accordingly, the importance of mobile GPUs is emerging and the most concern issue becomes a model for predicting the power and performance for smooth operation of high quality mobile contents. In many cases, the performance of mobile GPUs has been introduced in terms of power consumption of mobile GPUs using dynamic voltage and frequency scaling and throttling functions for power consumption and heat management. This paper introduces several studies of mobile GPU performance prediction model with user-friendly methods not like conventional power centric performance prediction models.

Key Words : Mobile GPU, Virtual Reality, Augmented Reality, Performance Prediction, Power Consumption

1. Introduction

Nowadays, the current 4G/LTE network mobile services are moving to 5G mobile communication era with the help of the advanced evolvement of information and communications technologies (ICT) and corresponding applications. 4G technologies may have limitations of providing services requiring both real-time response and big data sizes. Under 4G/LTE network, it is not possible for us to have instantaneous cloud services on mobile [1-3].

The new feature of 5G will include unlimited information exchange, massively diverse connectivity, and devices with self-sustained energy. Fig. 1 shows 5G service roadmap and network requirements made by 5G Forum [1]. This reports introduces the 5G mobile service scenario in which virtual reality/augmented reality service and massive content streaming service are introduced as examples of immersive 5G services.

This evolvement is on the base of the advanced development of smart devices and high quality graphics processing computing power of mobile application processors along with the development of network communication technology. Accordingly, the importance of mobile GPUs is emerging and the issues of low power and the performance for smooth operation of high quality mobile contents has become one of the most concern issues [4]. In many cases, the performance of mobile GPUs has been introduced in terms of power consumption of mobile GPUs using dynamic voltage and frequency scaling and throttling functions for power consumption and heat management [5-8].

This paper introduces several studies of mobile GPU performance prediction model with user-friendly methods not like conventional power centric performance prediction models. The rest of the paper is organized as follows. In Section 2, we review the basic concept of graphic processing pipelines and Qualcomm Adreno GPU module on Snapdragon. Section 3 introduce several studies of mobile GPU performance prediction model. Section 4 concludes this paper.

[†] E-mail: cgkim@nsu.ac.kr

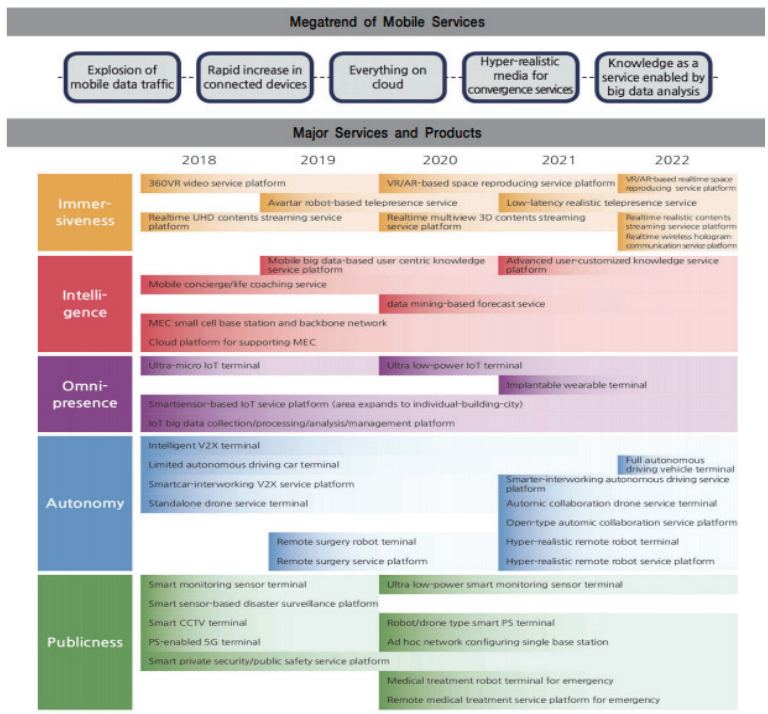

Fig. 1. 5G service roadmap and network requirements.

2. Background

2.1 Modern GPU

In general, the GPU receives geometry information from the CPU as an input and provides a picture as an output. Fig. 2 shows a diagram of modern graphic processing units. The host interface is the communication bridge between the CPU and the GPU. It receives commands from the CPU and also pulls geometry information from system memory. It outputs a stream of vertices in object space with all their associated information such as normals, texture coordinates, per vertex color, etc.

The vertex processing stage receives vertices from the host interface in object space and outputs them in screen space. This may be a simple linear transformation, or a complex operation involving morphing effects. They are are also transformed. No new vertices are created in this stage, and no vertices are discarded such that input/output has 1:1 mapping.

Next is triangle setup stage in which geometry information becomes raster information (screen space geometry is the input; pixels are the output). Prior to rasterization, triangles that are back-facing or are located outside the viewing frustum are rejected. Some GPUs also do some hidden surface removal at this stage. A fragment is generated if and only if its center is inside the triangle. Every fragment generated has its attributes computed to be the perspective correct interpolation of the three vertices that make up the triangle.

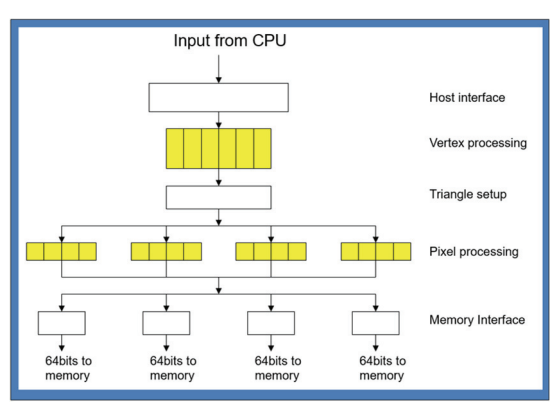

Fig. 2. Graphic processing pipelines.

Next is pixel processing stage. Each fragment provided by triangle setup is fed into fragment processing as a set of attributes (position, normal, texcoord, etc), which are used to compute the final color for this pixel. The computations taking place here include texture mapping and math operations. Typically, the bottleneck occurs here in modern applications.

Memory interface follows at last. Fragment colors provided by the previous stage are written to the framebuffer. Used to be the biggest bottleneck before fragment processing took over Before the final write occurs, some fragments are rejected by the zbuffer, stencil and alpha tests. On modern GPUs, z and color are compressed to reduce framebuffer bandwidth (but not size).

Memory interface follows at last. Fragment colors provided by the previous stage are written to the framebuffer. Used to be the biggest bottleneck before fragment processing took over Before the final write occurs, some fragments are rejected by the zbuffer, stencil and alpha tests. On modern GPUs, z and color are compressed to reduce framebuffer bandwidth (but not size).

As for the programmability in the GPU, vertex and fragment processing, and now triangle set-up, are programmable. The programmer can write programs that are executed for every vertex as well as for every fragment. This allows fully customizable geometry and shading effects that go well beyond the generic look and feel of older 3D applications. The graphics pipelines of a standard desktop GPU and a mobile GPU are almost identical.

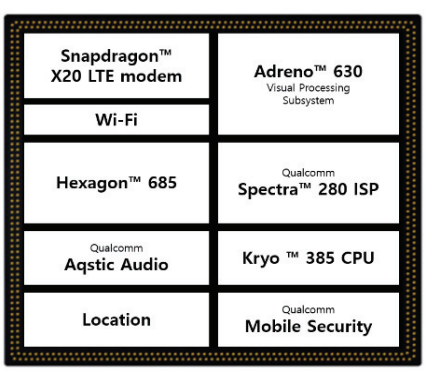

Fig. 3. Snapdragon platform.

2.2 Snapdragon 820 Mobile Processor

Fig. 3 shows the overview of Qualcomm's Snapdragon 820 processor [10], the leader in the mobile industry. It has a 64-bit dual core Kryo CPU that is enhanced approximately twice than its previous series of Krait CPU. It comes with well managed power management techniques, hyper threading, and automatic shutdown of caches, voltage scaling, frequency scaling, and other software level management techniques to optimize power and to increase performance of processors. It also equips with Hexagon-DSP to allow high quality multi-media processing with low power.

Graphics processor unit is essential for the display of smart phones for visualization. This allows user-friendly interfaces along with gaming and VR/AR application platforms. In addition to that, for the powerful camera features Snapdragon 820 makes use of Spectra ISP technology that support 14 bit intra spectra processor that supports three cameras.

3. GPU Performance Models

3.1 Adreno GPU Architecture

Generally, a mobile GPU has several processing units for graphics pipelines: unified shader (US), texture mapping unit (TMU), and render output processor (ROP). Fig. 4 shows the architecture of Adreno GPU and its graphics pipeline [9]. They are almost same as that of desktop GPU. Also, mobile GPUs equip with a programmable architecture in which the processing functions of vertex shader and fragment shader are integrated into. This architecture is called a unified shader and it performs the graphics pipelines consisting of vertex shader, geometry shader, pixel shader, tessellation, and computation operations. They have the functions of dynamic scheduling and load balancing systems. Because of that computing units can be allocated flexibly according to the amount of work to be handled [4].

TMU performs texture mapping and filtering operations cooperating with pixel and vertex shader units. ROP controls the sampling of the pixel; depth testing and alpha blending are its main tasks to determine the color of the final pixel. Finally, ROP writes all the rendered data to the frame buffer.

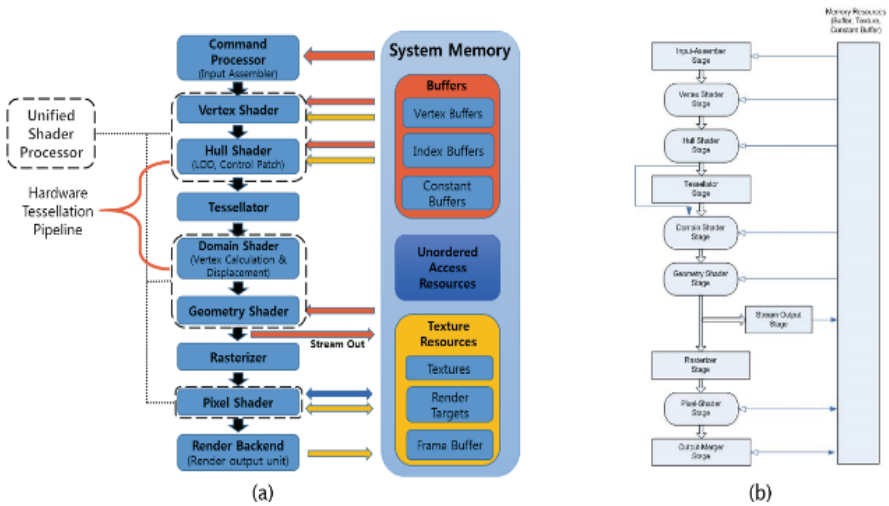

Fig. 4. Graphics pipeline of Adreno (a), Direct3D 10 graphics pipeline (b).

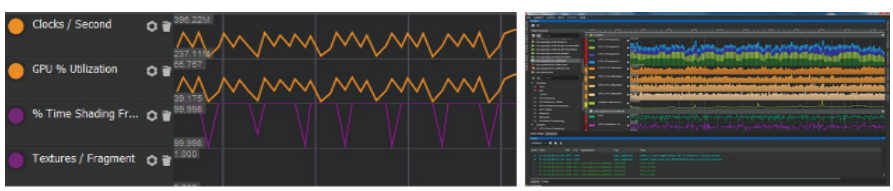

Fig. 5. Profiling analysis of Snapdragon.

Among these components, US comprises the greatest portion of a general GPU. Accordingly, almost all graphics processing tasks are handled by US. To understand the relationship between Adreno GPU H/W units and performance, the measurement of low-level performance value of the Adreno GPU using the snapdragon profiler as shown Fig. 5 was conducted [4].

3.2 GPU Performance Models

In mobile processors, the graphics processing performance directly affects the power consumption of the mobile GPU. From this point of view, there may be several methods of measuring and predicting power consumption of them. Several studies introduced the method of measuring the power consumption of each graphic processing step.

For this purpose, three-dimensional (3D) graphics scene attributes and pipeline primitive information sere used. In addition, a method of analyzing power consumption in the pipeline stage was introduces [11-13]. Huang et al [11] developed a micro-benchmark suite specifically tailored to the embedded GPU design based on graphic attributes and constructed a high-level energy model to assist graphic programmers to balance between performance, quality, and energy budget. Vatjus-Anttila et al [12] introduced a mathematical model for predicting power consumption. They measured the power performance according to the change of triangle based on the scene and camera view. Mochocki et al [13] analyzed the power models of 3D graphics pipelines in details by considering the effects of various 3D graphics factors such as resolution, frame rate, level of detail, lighting and texture maps on power consumption. In addition to that, this work identified and compared the benefits of candidate DVFS schemes for mobile 3D graphics pipelines. [14] analyzed the power consumption according to dynamic voltage and frequency scaling (DVFS) using the specifications of CPU and GPU monitoring the deviation caused by DVFS.

Xie et al. [6] proposed an estimation method for

Table 1. Mobile GPU performance models

Items	[4]	[11]	[12]	[13]	[14]
Platform	Mobile	Embedded	Mobile	Mobile	Mobile
Analytic Model	Mathematical equation	Linear regression	Mathematical model	None	Linear Regression
Model Factor	Instruction throughput on unified shader GPU frequency, the calculation speed of shaders (GFLOPS)	Energy consumption of each stage (Vertex Shader, Tile Accelerator, Image Synthesis Processor)	Triangles, Batches, Texels	Resolution, Frame rate, LOD, Lighting model, Texture Model	GPU Frequency & , Utilization

mobile GPUs. They used 3D rendering low-level performance which was measured at the early stage of SoC design. In addition, they built a linear regression model for estimating mobile GPUs. However, since the utilization factor which is variable due to DVFS is not considered, it may be difficult to predict performance close to actual performance.

A new method was introduced to predict mobile GPU performance by analyzing the relationship between hardware units that make up GPU performance. Ascertaining the maximum performance for which DVFS and throttling were considered [4]. Through micro-benchmarking, it measured low-level performance values and investigated the direct impact on performance by assigning workloads to each hardware device. On this basis of the working loads of US, the performance prediction model was made with the instruction throughput.

Table 1 shows several models for performance prediction mentioned before according to categories of characteristics. In the table, the "platform" identifies the device to be predicted, "analytic model" refers to the model used for prediction, and "model factor" refers to the factors used to derive the analytic models.

4. Conclusion

As the advanced development of smart devices and high quality graphics processing computing power of mobile application processors become popular, the importance of mobile GPUs is emerging. Therefore, one of the most concern issue is a model for predicting the power and performance for smooth operation of high quality mobile contents. Until now, the performance of mobile GPUs has been introduced mostly in terms of power consumption of mobile GPUs using dynamic voltage and frequency scaling and throttling functions for power consumption and heat management. This paper introduces several studies of mobile GPU performance prediction model with user-friendly methods not like conventional power centric performance prediction models.

References

1. 5G New Wave Towards Future Societies In The 2020S, 5G Forum(http://www.5gforum.org/), Mar. 2015.
2. Y. B. Park and Y. J. Kwon, "A Study on Dynamic Role-based Service Allocation for Service Oriented Architecture System," *Journal of the Semiconductor & Display Technology*, Vol. 17, No. 1, pp. 12-20, 2018.
3. Yong-Hwan Lee and Heung-Jun kim, "Evaluation of Feature Extraction and Matching Algorithms for the use of Mobile Application," *Journal of the Semiconductor & Display Technology*, Vol. 14 No. 4, pp. 56-60, 2015.
4. Juwon Yun, Jinyoung Lee, Cheong Ghil Kim, Yeong Kyu Lem, Jaeho Nah, Youngsik Kim and Woo Chan Park, "A Novel Performance Prediction Model for Mobile GPUs," *IEEE Access*, Vol. 6, pp. 16235-16245, 2018.
5. Y. G. Kim, M. Kim, J. M. Kim, M. Sung, and S. W. Chung, "A novel GPU power model for accurate

smartphone power breakdown," *ETRI Journal*, Vol. 37, No. 1, pp. 157–164, 2015.
6. Z. Xie, Y. Zhang, and L. Shi, "A method for estimating the 3D rendering performance of the SoC in the early design stage," *IEICE Electron. Exp.*, Vol. 11, No. 11, pp. 1–7, 2014.
7. A. Pathania, A. E.Irimiea, A. Prakash, and T. Mitra, "Power-performance modelling of mobile gaming workloads on heterogeneous MPSoCs," *in Proc. ACM/EDAC/IEEE DAC*, San Francisco, CA, USA, pp. 1–6. Jun. 2015.
8. C. Yoon, G. Ryu, and H. Cha. (Sep. 27, 2016). Utilization-based power modeling of modern mobile application processor. Yonsei University. [Online]. Available at http://mobed.yonsei.ac.kr/mobed_pages/pdf/mobedtr-2013-01.pdf
9. Microsoft. Graphics Pipeline. [Online]. Available at https://msdn.microsoft.com/en-us/library/windows/desktop/ff476882%28v=vs.85%29.aspx (accessed on 15 January 2017)
10. Francisco Cheng, "Snapdragon 820 – Technology and Traction," [Online]. Available at https://www.qualcomm.com/media/documents/files/
11. C. W. Huang, Y. A. Chung, P. S. Huang and S. L. Tsao, "High-Level Energy Consumption Model of Embedded Graphic Processors," *in Proc. IEEE DSP*, Singapore, Singapore, pp. 105-109, 2015.
12. J. M. Vatjus-Anttila, T. Koskela and S. Hickey, "Power Consumption Model of a Mobile GPU Based on Rendering Complexity," *in Proc. IEEE NGMAST*, Prague, Czech Republic, pp. 210-215, 2013.
13. B. Mochocki, K. Lahiri and S. Cadambi, "Power analysis of mobile 3D graphics," *in Proc. ACM DATE*, Belgium, Belgium, pp. 502-507, 2006.
14. Y. G. Kim, M. Kim, J. M. Kim, M. Sung and S. W. Chung, "A novel GPU power model for accurate smartphone power breakdown," *ETRI journal*, to be published.
15. C. Yoon, G. Ryu and H. Cha, "Utilization-based Power Modeling of Modern Mobile Application Processor," Yonsei Uni. [Online]. Available at http://mobed.yonsei.ac.kr/mobed_pages/pdf/mobed-tr-2013-01.pdf, Sep. 27, 2016.
16. Z. Xie, Y. Zhang and L. Shi, "A method for estimating the 3D rendering performance of the SoC in the early design stage," *IEICE Electronics Express*, Vol. 11, No. 11, p. 20140386, 2014.

접수일: 2019년 3월 23일, 심사일: 2019년 3월 25일, 게재확정일: 2019년 3월 25일