Vector Graph Implementations in E-Book Viewer Software and Cloud Platform

Hsin-Wen Wei[1], **Tseng-Yi Chen**[2], **Yen-Ju Lai**[2], **Yu-Yu Lai**[2], **Wei-Kuan Shih**[3]
[1]Deapartment of Information Management, TamKang University, Taipei, Taiwan
[2]Department of Computer Science, National Tsing Hua University, Hsinchu, Taiwan
[3]Institute of Information Systems and Applications, National Tsing Hua University, Hsinchu, Taiwan

Abstract - *Recently, many E-book readers have been developed to help people reading comic or novel on handheld devices. However, there are some challenges of developing e-book application such as image distortion, storage space, and vary of e-book format. Hence, we propose a system solution to solve these problems. We integrate vector graphic library into Android system and we also develop an e-book Android apps that combines cloud platform to extend e-book format supporting and user's storage space. Our solution achieves near-linear speedup on cloud platforms.*

Keywords: Vector graphic, e-book, cloud, android apps

1 Introduction

E-book became a popular application nowadays. There are three issues that we considered in this paper:

I. *Image distortion*: The contents and image of e-books may be distorted when user scales up the size of contents or images. Because of the image's format of e-book is the bitmap type.

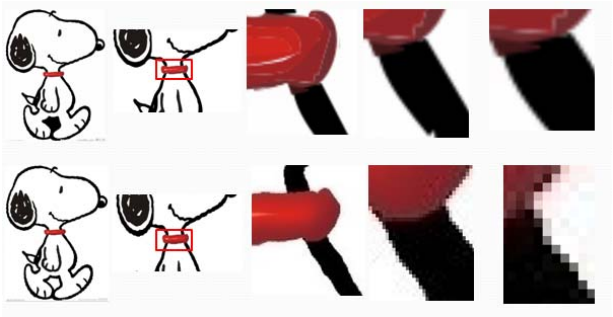


Figure 1. Bitmap and vector graphic comparison

II. *Storage management*: Users usually have many e-book contents and electronic comic books in their mobile devices. It is very inconvenient for user to manage their e-book content in small storage device. It occupies a large part of storage space in their device.

III. *Different e-book format*: There are many different e-book applications developed by different hardware manufacturers and software vendors. But there is no any e-book application or platform that can read any format of e-book contents.

To deal with these issues, we ported a well-known open source vector graphic library on Android system and developed an Android App. Then, we integrate our application into cloud platform. We take advantage of strong computing power of the cloud platform and cloud's large storage space to solve issue2 and 3.

The rest of this paper is organized as follows. Section II shows related work of vector graphic library and cloud image file transformation processing. Section III presents our system architecture in the cloud service. In Section IV, we present our experimental result and benchmark. The brief conclusion is presented in Section V.

2 Related work

E-book application has been widely studied by research group and industry manufacture in recent years. There are many different e-book file formats and most of them are not compatible. Therefore, it is an important issue of uniform format for e-book application. Vector drawings can enlarge to any size without any loss in quality. In addition, vector graphic is a free file format and it is also readable for any handheld devices and computers. So that, we can convert e-book file to xml descriptor of vector graphic. There are many libraries to support vector graphic reader in embedded platform [1] [2]. We integrate OpenVG [1] library with Android system and we also optimize it on our embedded platform.

In [3], the authors proposed some algorithms for converting file to vector graph format. They only deploy client server architecture to execute their algorithm. However, converting file format produce a huge workload when user need to transfer large amount of files. Our solution is a parallel implementation of the Autotrace that uses multi-VM

and it achieves near-linear speedup on cloud computing platform.

Another advantage of cloud environment is the huge storage space [4]. We store e-book files in cloud storage. After authorization, users can access their contents. If users are publishers, they can authorize their files to readers.

3 System infrastructure

As shown in Figure 2, there are three parts of our system: vector graphic library in end device, cloud platform for e-book file format transformation and cloud platform for file access and storage management.

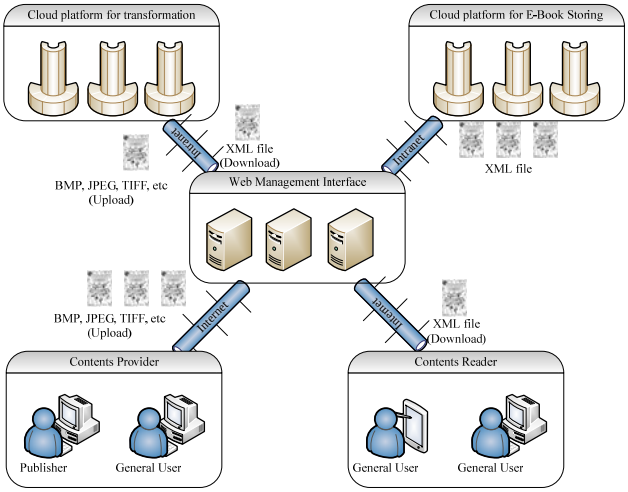


Figure 2. Our E-book reader system with cloud platform

3.1 Vector graphic library in end device

We modify the vector graphic library based on ShivaVG and port it on Android system. ShivaVG is a vector graphic library that follows OpenVG standard for library implementation. OpenVG separates the procedures of drawing vector graphic to seven steps. We optimize the vector graphic library in step four on Android system.

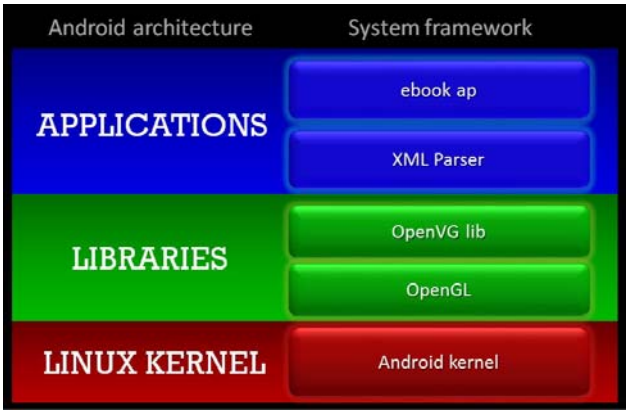


Figure 3. Vector graphic library on Android

Triangulation is the main process in rasterization step. The time complexity is $O(n^3)$ in original algorithm. We replace original triangulation algorithm with monotone triangulation and polygon partitioning. In the result, the time complexity of triangulation reduce from $O(n^3)$ to $O(n*\log n)$.

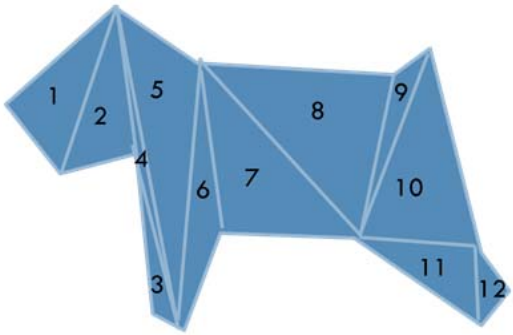


Figure 4. Example of monotone triangulation

There are some problems in monotone triangulation. If the picture is not a monotone graphic, monotone triangulation will get error in execution. So, we deploy polygon partitioning before monotone triangulation. Polygon partitioning separates any graph shape into monotone graph.

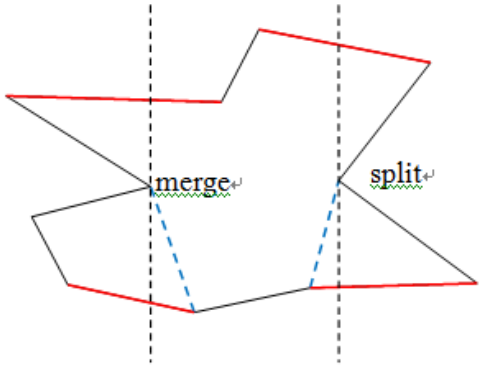


Figure 5. Example of polygon partitioning

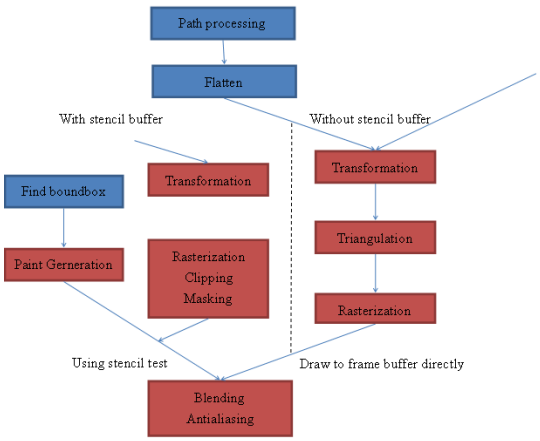


Figure 6. Flow of vector graphic library

3.2 Cloud platform environment

This research work converts wide variety of image format to vector graphic descriptor in order to support more and more file format. However, file transformation procedure is a heavy workload process in embedded platform. For that reason, our solution takes advantage of cloud computing in order to enhance our computing power.

We deploy the Autotrace [5] code in cloud. Autotrace is software of file transformation. And we build up a domination machine to dispatch converting task to back end cloud platform. We parallelize the file format conversion task in group of images method. Our solution splits all images which are in e-book file to different group and the system assigns these group tasks to different computing virtual machine in cloud. We present our cloud solution in figure 7.

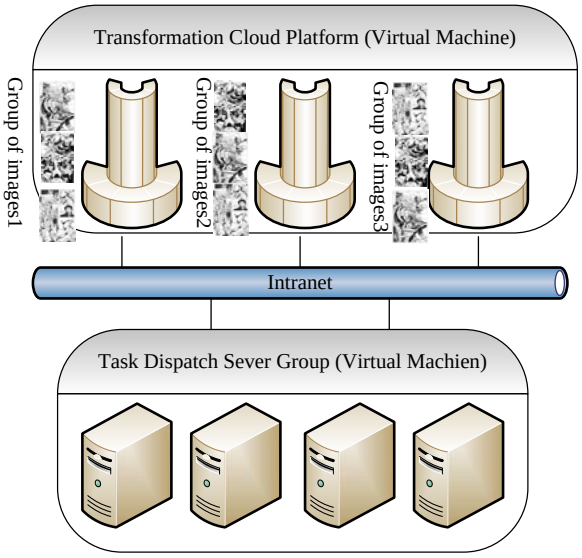
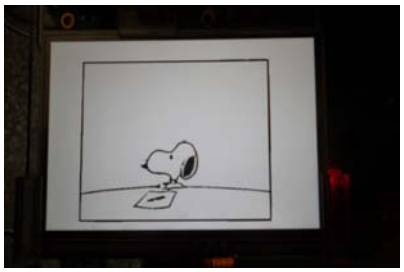


Figure 7. Cloud Parallel Method

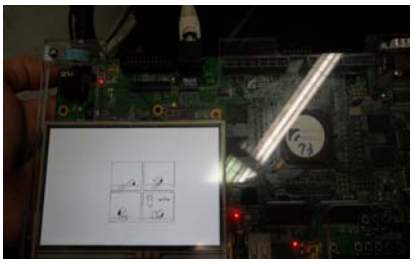
Users can manage their contents through the designed web interface. The management operations that can be performed on the website include uploading, downloading, and sharing their contents with copyright. We also put all contents which are uploaded by the same user in single cloud storage. The placement method can increase data locality.

4 Experimental Result

Our experimental environment is described as follows: smart phone, android pad and embedded development platform (PAC Duo), and Hicloud [6] computing platform. We show our result in figure 8 and performance result in figure 9.



(a) E-book application on PAC Duo platform



(b) Browser e-book contents on PAC Duo platform

Figure 8. E-book application on PAC Duo platform

Figure 8 shows the result of our system execution on embedded platform. Then, figure 9 presents the gaining performance when the number of virtual machine increased. Our solution uses multi-VM and achieves near-linear speedup on cloud computing platforms.

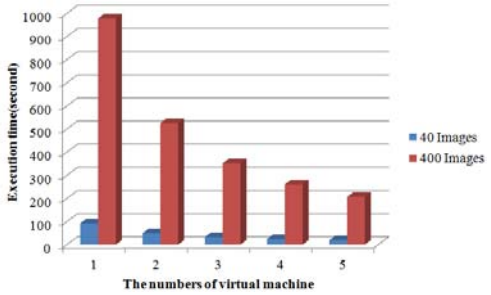


Figure 9. Performance of file transformation

5 Conclusion

We propose a total solution for e-book application, including: system library, cloud file transformation and cloud storage. Our work also achieves near-linear speedup by parallelizing process of file transformation on cloud computing platform. In addition, we also increase performance of local vector graphic library on Android system. The optimization result of vector graphic reduce the time complexity from $O(n^3)$ to $O(n*\log n)$. Finally, our application system is ready to market. It will be a popular application system.

6 Acknowledgement

The work reported in this paper was supported by National Science Council under grant NSC101-2218-E-032-003 and NSC100-2221-E-007-103.

7 References

[1] Ferraiolo, J. “Scalable vector graphics (SVG) 1.0 specification”. in Tech. rep., W3C Recommendation. Sep, 2010.

[2] Aekyung Oh, Hyunchan Sung, Hwanyong Lee, Kujin Kim and Nakhoon Baek, “Implementation of OpenVG 1.0 using OpenGL ES”. In MobileHCI’07, 2007.

[3] Zhimao Guo, Shuigeng Zhou, Zhengchuan Xu, and Aoying Zhou. “G2ST: a novel method to transform GML to SVG”. in ACM GIS’03, 2003

[4] Ann-Marie Horcher and Maxine Cohen. “Ebook Readers: An iPod for Your Books in the Cloud”. In Communications in Computer and Information Science, Vol. No.174, Pages 22-27, 2011.

[5] AutoTrace converts bitmap to vector graphic, http://autotrace.sourceforge.net/

[6] CHT Hicloud, http://hicloud.hinet.net/