Chapter 14
On Balancing Energy Consumption, Rendering Speed, and Image Quality on Mobile Devices

Fan Wu
Tuskegee University, USA

Emmanuel Agu
Worcester Polytechnic Institute, USA

Clifford Lindsay
Worcester Polytechnic Institute, USA

Chung-han Chen
Tuskegee University, USA

ABSTRACT

Mobile games and graphics are popular because untethered computing is convenient and ubiquitous entertainment is compelling. However, rendering graphics on mobile devices faces challenges due to limited system resources, such as battery energy, and low memory and disk space. Real time frame rates, low energy consumption and high image quality are all desirable attributes of interactive mobile graphics; however, achieving these objectives is conflicting. For instance, increasing mesh resolutions improves rendered image quality but consumes more battery energy. Therefore, the authors propose a mobile graphics heuristic to minimize energy consumption while maintaining acceptable image quality and interactive frame rates. Over the lifetime of a mobile graphics application, scene complexity, animation paths, user interactivity and other elements all change its CPU and resource demands. In this regard, a heuristic that dynamically changes scene mesh LoDs and amount of CPU timeslices allotted to the mobile graphics application is presented to select optimal operating conditions that balance rendering speed, energy conservation and image quality. Additionally, a workload predict model is proposed so that the heuristic can monitor both application workload and the availability of resources of mobile devices periodically, while adaptively determining how much resources will be allocated to applications.

DOI: 10.4018/978-1-4666-0921-1.ch014
INTRODUCTION

Battery-powered mobile devices, ranging from laptops to cell phones, have become important platforms for running 3D graphics applications. Traditionally, 3D graphics applications have been developed exclusively for desktop computers or dedicated gaming consoles. However, with improved mobile processing power, screen resolution and programmable graphics hardware, applications are increasingly being targeted at mobile platforms. For example, online multiplayer games, mobile telesurgery and other mobile graphics are becoming more prominent. Commercially, 3D graphics applications offer a new opportunity especially considering that annually the total number of mobile devices sold far exceeds the number of personal computers sold. The mobile gaming industry already reports revenues in excess of $2.6 billion worldwide annually, and is expected to exceed $11 billion by the year 2010 (Mobile Games, 2005).

The most prominent factor limiting the usefulness for mobile devices is short battery life. In fact, while mobile CPU power, memory and disk space availability has grown exponentially over the years, battery capacity has only increased 3-fold in the past decade. Consequently, the mobile user is frequently forced to interrupt their mobile graphics experience to recharge dead batteries. Therefore, application directed energy saving techniques have become an active topic within the mobile computing community.

The main idea is that during periods when a mobile application’s workload reduces, energy can be saved by scheduling fewer CPU timeslices for it, or the CPU clock speed can be reduced (Dynamic Voltage and Frequency Scaling (DVFS)). Our main thrust in this project is to introduce application-directed energy saving techniques into mobile graphics applications. However, scheduling less CPU cycles for a real-time graphics application causes a reduction in frame rate. Also, the fraction of CPU cycles that is required to run a given application at 25 frames per second depends on the scene complexity and Level-of-Detail (LoD). Essentially, energy efficiency, frame rate, and visual realism have complex inter-relationships. Thus, CPU scheduling must be integrated with LoD selection.

The workload of most real-time graphics applications varies over its lifetime due to changes in many factors such as user interactivity, the Level-of-Detail (LoD) of scene models, and the complexity of animation and lighting. If a fixed fraction of the CPU’s cycles are used to run such a real time application (no dynamic scheduling), its frame rate fluctuates over time. Apart from being annoying to application users, a fluctuating frame rate causes wasteful spikes in mobile energy usage. Ideally, a constant frame rate of 25 FPS would produce an ideal energy usage scenario. In some of our measurements, we found that such spikes in frame rate (and energy consumption) above our target real-time frame rate can consume up to 70% more energy as shown in Figure 1.

In many cases, techniques that optimize one of these attributes can degrade another. For instance, improving image quality requires increasing mesh Level-of-Detail (LoD), which requires more CPU cycles to render and unnecessarily consumes energy from the mobile device’s battery. These complex interdependences mean that CPU scheduling has to be coupled with LoD management to balance the three desirable attributes of mobile graphics: photorealism, real-time rendering and energy efficiency.

Our integrated heuristic attempts to select the minimum LoD that i) yields acceptable visual realism, and schedules just enough CPU timeslices to ii) maintain real-time frame rates (25 Frames Per Second (FPS)) such that iii) the energy consumed by the mobile graphics application is minimized. To the best of our knowledge, our proposed strategy is the first attempt to employ CPU scheduling to save energy in mobile graphics. We propose a comprehensive algorithm that statistically estimates the CPU cycle requirements...
Related Content

Prediction and Validation Approach with Accumulation Applied to Video Segmentation
www.igi-global.com/article/prediction-validation-approach-accumulation-applied/66363?camid=4v1a

Service Provision for Pervasive Computing Environments
www.igi-global.com/chapter/service-provision-pervasive-computing-environments/17189?camid=4v1a

Security Architectures for B3G Mobile Networks
www.igi-global.com/chapter/security-architectures-b3g-mobile-networks/26685?camid=4v1a

GSM-Based Positioning Technique Using Relative Received Signal Strength
www.igi-global.com/article/gsm-based-positioning-technique-using-relative-received-signal-strength/103152?camid=4v1a