Ray tracing

Ray tracing is a rendering technique that models the physics of light to generate realistic images. In many modern processors, the necessary computations utilize a floating-point representation to approximate real numbers. Floating-point numbers capture a very large range of numeric values, but their precision decreases as magnitude increases. This issue can sometimes lead to visible artifacts in the image. Moreover, some high-performance processors do not provide hardware support for floating-point values. As an alternative, a fixed-point representation based on integer values can be used to capture real numbers precisely, but only in a limited range. Converting the standard ray tracing algorithm to utilize a fixed-point representation presents many interesting and challenging problems, and we have explored existing answers, as well as created our own novel solutions, to these fixed-point difficulties.

Integer-only ray tracing pipeline

We have implemented a complete, full-featured integer ray tracing pipeline targeting computational environments that lack hardware support for the IEEE 754 floating-point standard. Whereas hardware support for this standard is nearly universal among common general-purpose central processing units (CPUs) and modern graphics processing units (GPUs), many processors—such as low-power embedded processors, high-performance stream processors, specialized digital signal processors, and so forth—do not provide hardware support for IEEE 754. Moreover, eliminating floating-point units in the design of a custom ray tracing architecture can significantly reduce power and area requirements, but still deliver performance competitive with processors that provide full hardware support for the standard.

Although it would be easy to dismiss the problem of integer ray tracing as a matter of abstraction—that is, one solved simply by designing an appropriate fixed-point data type for use in place of standard floating-point types—our experience shows that a correct, efficient, and accurate solution actually involves a number of subtle details that complicate this view: for better or worse, even a well-designed fixed-point data type cannot serve as a simple drop-in replacement for the standard IEEE 754 floating-point types.

Target architecture: SPI Storm-1 series processors

Stream Processors, Inc. (SPI) has developed a stream processor to target embedded media and signal processing applications with specific attention to low cost, power efficiency, and programming simplicity. SPI’s hardware architecture is based on a standard RISC processor that offloads performance-critical tasks, such as ray tracing computations, to an integer-only, data-parallel processing unit designed to efficiently exploit the levels of parallelism found in many applications.

Conventional multicore CPUs that use coherent caches are inefficient to support intense computation with little data reuse. In contrast, the SPI architecture uses a sophisticated C compiler to manage DMA to transfer only the data needed. The efficient data management results in an order of magnitude improved compute performance and power-efficiency without increasing cost or development effort.

Current results

We can observe that modern general-purpose CPUs, such as the 3.06 GHz Intel Core2 Duo used to generate these data, are highly optimized for floating-point performance: the floating-point ray tracer executes approximately 3x-4x faster than the integer-only ray tracer. However, we expect significant performance improvements once implementation for the SPI Storm-1 stream processor is complete.

Nevertheless, the image quality of the integer-only ray tracing pipeline is at least as good as, and in some cases better than, that of the floating-point ray tracer.

Acknowledgements

This work was funded by the Swezey Scientific Instrumentation and Research Fund at Grove City College. The authors gratefully acknowledge the assistance of Johannes Hanika at Universität Ulm in Germany and Dr. Ujval Kapasi, Founder & Director of Software Applications at Stream Processors, Inc.