Memory System Support for Image Processing

Lixin Zhang, John B. Carter, Wilson C. Hsieh, Sally A. McKee

UUCS-99-002

School of Computing
University of Utah
Salt Lake City, UT 84112 USA

October 12, 1999

Abstract

Image processing applications tend to access their data non-sequentially and reuse that data infrequently. As a result, they tend to perform poorly on conventional memory systems due to high cache and TLB miss rates and are particularly sensitive to the growing latency of main memory. In this paper, we analyze the memory performance of three image processing algorithms (*volume rendering*, *image warping*, and *image filtering*) on both a conventional memory system and on the Impulse memory system. The Impulse memory system allows application software to control how, when, and where data are loaded into a conventional processor cache. It does this by letting software configure how the memory controller interprets the physical addresses exported by the processor, which enables an application to dynamically change how data are fetched. Sparse data can be accessed densely, which improves both cache and TLB utilization, and memory latency is hidden by prefetching data within the memory controller. We find that for these image processing codes, using an Impulse memory system yields speedups of 40% to 226% over an otherwise identical machine with a conventional memory system.