Accelerating Software-based MPEG Encoding Using the VIS^{TM} Instruction Set

John E. Stone
Heuris
St. Louis, MO

Ahmad Zandi
Sun Microelectronics
Mountain View, CA

Abstract

MPEG is a set of standards for the compression and coding of digital audio and video signals. In order to provide high compression ratios without loss of quality, MPEG encoders perform complex analyses on source material. These analyses require billions of operations to encode even a short sequence of video lasting only seconds. Even the fastest general-purpose microprocessors available today lack the speed to provide real-time MPEG encoding in software. For applications that absolutely require real-time operation, the only solution is to use dedicated hardware. Hardware-based MPEG encoders typically suffer from the same problems that most special purpose machinery does— high cost, little flexibility, and long product development cycles. Fortunately most MPEG encoding applications do not require real-time operation. Software encoders trade speed for low cost, portability, upgradeability, and flexibility. The growing use of general purpose computers in multimedia applications has led leading computer manufacturers to add multimedia acceleration instructions to their latest processor designs. Through the use of multimedia acceleration instructions, software-based MPEG encoders gain a tremendous amount of speed, at low cost, without sacrificing their advantages.

1. Introduction

Most commercial MPEG encoding products available today are hardware-based solutions. These encoders come in the form of stand-alone dedicated systems, and as expansion boards for general-purpose computers. Dedicated, hardware-based MPEG encoders have several drawbacks. Hardware is expensive to design and expensive to upgrade. Most MPEG encoding hardware is designed to operate in real-time and is constrained by a limited set of features that vary proportionally with cost. In order to perform the billions of operations necessary to produce high quality video, hardware solutions require special-purpose VLSI chips, which are costly to design and manufacture due to low production volume. Product design cycles for custom hardware are slow compared to that for software products. Due to the constraints that hardware encoders must accomodate, they usually acquire source material by digitizing analog inputs on-the-fly while encoding. This is a significant drawback to flexibility, as it requires that all video be re-digitized regardless of its source format. Now that digital editing systems are gaining widespread use, analog input requirements are becoming a hindrance. Each time signals are converted between digital and analog representations, a small amount of unavoidable signal degradation occurs.

Software-based MPEG encoding solutions provide much more flexibility and cost much less than their hardware counterparts. Software products have a much shorter development cycle than hardware, allowing earlier availability of advanced features. They can be upgraded quickly and inexpensively, making it easy for customers to stay up-to-date with current technology. Software encoders are not normally designed for real-time encoding, so they are free to employ sophisticated algorithms that are too complex or expensive to implement in hardware. They can also be easily integrated with digital editing systems to form a completely digital content production system. Most hardware solutions require analog video input, but software encoders can encode digital media files directly from editing systems with no loss in quality. Through the use of editing information, multi-pass encoding, and advanced image processing algorithms, software encoders can produce higher quality output than most real-time hardware solutions. Heuris has developed software-based MPEG encoding solutions that overcome many of the limitations associated with hardware encoders.

Until recently, the only way to accelerate multimedia software was with special-purpose processors specifically designed for digital signal processing— DSPs. Sun's UltraSPARC processors are the first to implement a comprehensive set of multimedia extensions called the Visual Instruction Set[2] to accelerate multimedia applications. Sun's introduction of VIS-enhanced microprocessors leads an industry trend of adding specialized instruction sets which diminish the need for special-purpose DSP-acceleration to run data-intensive multimedia applications.

The VIS instruction set accelerates operations in image processing[3], real-time audio/video processing, and 3-D rendering, among others. Because UltraSPARC employs 64-bit data paths, the VIS instruction set can operate on eight 8-bit, four 16-bit, or two 32-bit operands concurrently. This can provide an immediate 8x, 4x, or 2x speedup over software using conventional arithmetic instructions. Since the VIS instructions are implemented on the UltraSPARC processor[1] itself, processing speed scales with clock rates (167MHz, 200MHz and higher) and with the number of processors in a multiprocessor architecture. The scalability offered by this approach makes a compelling argument for doing MPEG encoding on multimedia-enhanced microprocessors.

VIS technology preserves all of the benefits of software-based encoding and additionally provides many of the benefits of specialized acceleration hardware. Software modules can be VIS-enhanced one at a time, providing speed increases which make an immediate impact while preserving the existing code base. This paper highlights the benefits of software-based MPEG encoding and will illustrate the performance improvements provided by the VIS instruction set on UltraSPARC platforms.

2. Overview of MPEG Video Encoding

In order to understand how technologies such as VIS may be harnessed to accelerate MPEG compression, it is best first to examine what kinds of operations MPEG encoding consists of. For the purposes of this paper, we will limit the scope of discussion to MPEG video encoding.

MPEG video encoders perform billions of calculations while processing video, operating on small blocks of pixels at a time. One way to think about this process is to think of a grid drawn over a television screen. The blocks in this grid break an image up into small manageable pieces that are convenient to perform arithmetic operations on. In the course of encoding video, a repetitive set of operations is applied to the blocks in each image as it is encoded. Encoding operations include scaling, color space conversion, motion estimation, noise filtering, transformation from spatial representation to frequency representation, quantization, interpolation, and entropy encoding. As each frame in a video sequence is encoded, operations are applied to either the entire image at once or one block at a time.

MPEG video encoding operations require a large number of memory accesses. For each pixel processed in motion estimation, scaling, and filtering, many neighboring pixels are referenced. Motion estimation operations potentially reference pixels in more than one frame at a time. Due to the data-intensive nature of video encoding, memory performance directly affects encoding speed. Hardware encoders employ high-speed memory and special-purpose circuitry to optimize these block operations and memory accesses. General-purpose computers typically employ multi-level memory systems that consist of relatively small high-speed memory caches, and larger, but lower-speed, main memory. Another attribute of general-purpose computers is that they are typically designed to work on pairs of data elements at a time. Most computer applications operate on only one or two data elements at a time. MPEG encoding, 3-D rendering, and other multimedia applications benefit greatly from the ability to perform the same operation on many data elements at once.

Hardware encoders have the benefit of being designed for a single purpose. They implement memory systems and circuitry that are ideal for the task at hand. Hardware encoders typically implement circuitry that performs operations on many pixels at a time. This specialization, which is advantageous for performance, limits the flexibility of hardware encoders and raises product cost.

For most MPEG video encoding needs, real-time operation is not a requirement. Software encoders can be designed to accept a wide range of input, allow detailed control over the encoding process, and include features that are not feasible for implementation in fixed functionality circuitry. Although flexible and cost-effective, software encoders are subject to the limitations of the computers on which they run. The most common issues faced in MPEG video encoding are ease of use, flexibility, encoded video quality, and speed of encoding. Software encoders can outclass hardware solutions in all areas but speed. Many microprocessor vendors are starting to add multimedia acceleration features to general-purpose microprocessors found in desktop computers. Software encoders can take advantage of available multimedia acceleration features to narrow the speed gap with hardware encoding solutions. An important aspect of this strategy is that software encoders retain their advanced functionality, ease of use, and general flexibility, while gaining in performance. With the use of multimedia extensions such as VIS, none of the advantages of software encoding are compromised.

3. VIS Acceleration

In the previous sections we have outlined some operations that take place in MPEG video encoding, and how these operations affect the design and performance of both software and hardware video encoders. Sun's VIS instruction set extension to the UltraSPARC processor provides many of the capabilities found in dedicated hardware encoders but on a smaller scale. The VIS instruction set adds the ability to perform arithmetic operations on many image pixels at once— up to eight at a time. By operating on several pixels simultaneously, the UltraSPARC processor significantly reduces the total number of instructions executed in the course of video encoding.

A simple example of a VIS performance gain is the addition of two groups of four pixels. A standard unaccelerated processor can perform such an addition by operating on four pairs of pixels, one pair at a time. An UltraSPARC processor may perform the same operation in approximately one fourth the time by doing all four additions simultaneously. This approach is similar to what hardware MPEG encoders do to increase their performance. Because VIS instructions operate on groups of pixels at a time, the memory system is used in an efficient manner. Since many of the operations involved in MPEG encoding are as data-intensive as they are arithmetic-intensive, the ability to perform identical operations on many pixels at once provides an immediate performance gain.

In order to implement VIS-enhanced processing, one needs only to replace traditional pixel-by-pixel processing loops with loops of VIS instructions that operate on several data elements at a time[3]. One great feature of multimedia extensions like VIS is the ability to enhance software one module at a time, accelerating operations that need it the most first. This method proved to be very successful in the initial stages of developing a VIS-enhanced MPEG encoder. As development of VIS-enhanced software continues, a programmer may gradually enhance modules one at a time, in order of decreasing priority. One of the key areas to be enhanced in an MPEG encoder is motion estimation. Of all the operations involved in MPEG video encoding, motion estimation is the most time-consuming. Motion estimation involves comparing the level of similarity between blocks of pixels in multiple images and different locations within the images. This process is very time-consuming because it involves a tremendous amount of memory references as well as arithmetic operations. VIS helps dramatically with this process by supplying key instructions for this very purpose. The PDIST[2] instruction computes the absolute value of the difference between two 8 pixel pairs. It accomplishes the work of approximately 48 standard arithmetic instructions in a single instruction.

4. Initial Performance Results

Initial experiments with implementing a VIS-accelerated video encoder based on the Heuris MPEG Power Professional software have been very encouraging. Since most of the computation is spent on motion estimation calculations, those routines were the first to be accelerated. Single processor timing tests were conducted on a Sun Ultra 1/170E. Multiprocessor tests were performed on a Sun Ultra 2/2170. Before VIS-acceleration, average encode rates for default settings of the encoder were running at a ratio of 45:1 versus real-time. At a ratio of 45:1, the original unaccelerated encoder took 45 seconds to encode each second of video.

Early attempts to accelerate the motion estimation code with VIS instructions yielded an immediate 2x speedup over the original. As the implementation was refined, the VIS-accelerated motion estimation code achieved a 6x increase in speed. Although this speed increase is very significant, motion estimation is only one part of the MPEG encoding process. With the tremendous speedup achieved in the motion estimation code, other encoder operations began to consume proportionally larger percentages of the total computation time. After spending a short time optimizing a small number of additional functions, the experimental encoder now achieves a ratio of 15:1 versus real-time. As an end result, the experimental MPEG encoder runs a factor of three times faster than the original, with minimal rewriting. A current work in progress based on the use of VIS acceleration and the use of two processors has already achieved performance as high as 10:1 versus real-time, in its earliest stages. Several time-consuming operations within the MPEG encoder remain that when VIS-enhanced, will significantly increase overall performance. It is expected that a performance ratio of better than 5:1 versus real-time will be achieved when the remaining functions in the encoder are accelerated. These performance results clearly indicate that the VIS instruction set on the UltraSPARC processor provides spectacular performance benefits in MPEG encoding software.

References

[1] UltraSPARC-I User's Manual, January 1996. Sun Microelectronics.

[2] Visual Instruction Set User's Guide, April 1996. Sun Microelectronics.

[3] Daniel S. Rice. High-Performance Image Processing Using Special-Purpose CPU Instructions: The UltraSPARC Visual Instruction Set. Master's thesis, University of California, Berkeley, April 1996.