



FAU Institutional Repository

<http://purl.fcla.edu/fau/fauir>

This paper was submitted by the faculty of [FAU's Harbor Branch Oceanographic Institute](#).

Notice: ©1998 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

This manuscript is available at <http://ieeexplore.ieee.org/> and may be cited as: Kocak, D. M., & Caimi, F. M. (1998). DSP hardware implementation of transform-based compression algorithm for AUV telemetry. *Oceans '98: Conference proceedings: 28 September-1 October, 1998, Nice, France, Acropolis Convention Center*. (Vol. 3, pp. 1624-1628). Piscataway, NJ: Oceans '98 IEEE/OES Conference Organizing Committee. doi:10.1109/OCEANS.1998.726363

DSP HARDWARE IMPLEMENTATION OF TRANSFORM-BASED COMPRESSION ALGORITHM FOR AUV TELEMETRY

Donna M. Kocak
kocak@hboi.edu

Frank M. Caimi
caimi@hboi.edu

Department of Electrical and Software Engineering
Harbor Branch Oceanographic Institution, Inc.
5600 U.S. 1 North
Ft. Pierce, FL 34946

Abstract: Operation of AUVs or other platforms that rely upon robust communications for navigation and mission critical tasks is facilitated by the use of on-board data compression. Data compression eliminates redundancy of transmitted information and therefore improves the efficiency of the entire mission. In some applications, acoustic and non-acoustic sensor information is acquired and processed to produce a visual representation for ease of human-in-the-loop analysis and interpretation.

Many image and data compression methods are currently in development. Emphasis has been placed on achieving high efficiency as characterized by compression ratio while retaining the least error between the original image and the compressed image. Although some algorithms produce excellent results, computational complexity, and hence speed of compression or decompression, can become a limiting factor in real-time use depending upon the capability of the on-board processor.

This paper describes the implementation of a high performance image compression transform using a dedicated DSP processor and supporting computer host. The system architecture and strategy for efficient code implementation are described. Example performance measures are presented for non-acoustic undersea. Future plans and developments for specialized applications are also discussed.

I. INTRODUCTION

Data communications between an underwater AUV or other platform and a topside host is often essential for navigation and mission critical tasks. In applications such as AUV image-based underwater surveillance or automated target recognition (ATR), either raw images or preprocessed image data must

be transmitted over this link. These applications pose several implementation challenges. One obvious challenge is data throughput due to the nature of the intended applications. For human-in-the-loop target cueing, image data must be analyzed and viewed as quickly as possible (usually from several to 30 fps for RS-170/NTSC video). In order to view regions of interest in a captured image, image resolution must be moderate to high, thus requiring a large amount of data. This can severely limit transmission time since generally low-bandwidth uplinks such as acoustic modems are used. To facilitate transmission, on-board data compression techniques are implemented. Data compression eliminates redundancy of transmitted data and therefore improves the efficiency of the entire mission. Many image and data compression methods are currently in development. Emphasis has been placed on achieving high efficiency as characterized by compression ratio (CR) (defined as data volume input to output) while retaining the least mean squared error (MSE) between the original (input) image and the compressed (output) image. Although some algorithms produce excellent results, computational complexity, and hence speed of compression or decompression, can become a limiting factor in real-time use depending upon the capability of the on-board processor. Other factors limiting data compression include camera (image intensifier) noise and media point-spread-function (loss of detail). Early in

our ONR sponsored program, HBOI and University of Florida (UF) evaluated several common data compression transforms for underwater imaging [1]. Although a few methods were found to be useful for low and moderate CRs (VPIC and SPIHT < 50:1; wavelet < 70:1; EPIC < 150:1), we have developed the high-compression transform BLAST (CR > 240:1) [1,2]. A comparison of CR vs. MSE for common compression methods is shown in Figure 1.

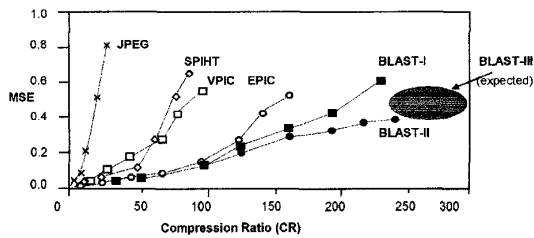


Figure 1. Comparison of Common Data Compression Methods.

This paper describes the implementation of the BLAST transform using a dedicated DSP processor and supporting computer host. The system architecture and strategy for efficient code implementation are described. Example performance measures are presented for non-acoustic undersea. Future plans and developments for specialized applications are also discussed.

II. SYSTEM DESCRIPTION

The system is designed for use on an AUV but could easily mount on a submersible or other underwater platform (refer to Figure 2). There are three major components: system hardware, transform-based compression algorithm and host computer program. Each of these components are discussed in the following sections.

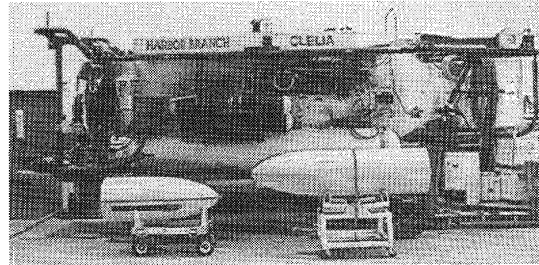


Figure 2. HBOI Clelia submersible and AUV payload sections (3-D side scan sonar (left) and Compressive Underwater Video Camera (CUVC) (right)).

A. System Hardware

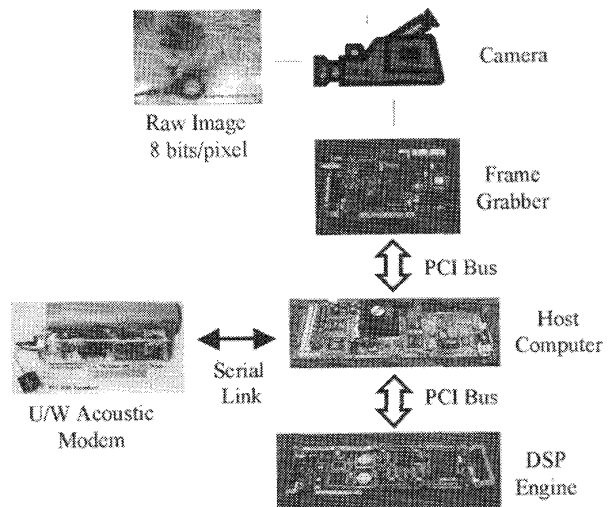


Figure 3. System configuration showing data flow (FAU acoustic modem shown).

High processing speeds, flexibility, and expandability were the primary factors in the selection of system components. We have taken a bottom-up approach to development in which tasks are first accomplished at a basic level, tested, and then optimized for improved performance. The current system configuration is shown in Figure 3. With the large number of data transfers required in passing information, overall system performance is influenced by the I/O bus. Thus, to achieve the best performance in bus speed, peak throughput, and sustained throughput of 8-bit pixel data to system

memory, a PCI interface bus was chosen over the competing ISA, EISA and NuBus choices (summarized Table 1).

	PCI	ISA	EISA	NUBUS
Bus Speed (MHz)	33	8	8	10
Theoretical Peak Throughput (MB/s)	132	8	32	40
Sustained Throughput of 8-Bit Data to System Memory (MB/s)	80 w/ DMA	1.8 w/ DMA	10 w/ DMA	8 w/ DMA

Table 1. Interface Bus Specifications [3].

The frame grabber, host computer, and DSP boards connect to the PCI bus. RS-170/NTSC images are acquired from a Sony DCR-VX1000 digital video camera with photo mode capability. (A digital still camera system is being developed concurrently.) A National Instrument's IMAQ PCI-1408 frame grabber board uses an antichrominance filter to provide full 8-bit accuracy monochrome images (raw format) and provides various triggering features (4 external lines, RTSI bus, and programmable modes) and speed enhancers (programmable ROI, hardware scaling, LUTs, and MITE data transfer ASIC). Using master-mode DMA the acquisition board transfers up to 132 MB/s of raw image data to the host computer. Once receiving an image, the host computer sends the raw image to the DSP engine, waits for the compressed image output, then sends the compressed image to the acoustic modem (~8 Kb/s serial port interface) for transmission to a topside modem [4]. Eventually, a networked communications system is anticipated using multiple AUVs [5]. The host computer is a Pentium Pro-200 industrial single-board computer (SBC) equipped with Windows NT operating system. Windows NT affords multiprocessor capability, complete system protection, flexible development platform, and high reliability and stability. Furthermore, NT increases performance of the DSP engine over Windows 95 by a factor of 10 by using DMA bus transfers. The DSP

engine is comprised of an Analog Devices SHARC 3000 processor resident as part of a motherboard / SHARCPAC configuration manufactured by Alex Computers. The SHARCPAC is a modular mezzanine card based on the Analog Devices 2106x SHARC DSP chip. The SHARCPAC / motherboard operates in a standard IBM PC backplane configuration and communicates via standard PCI bus architecture. SHARCPAC modules host up to eight SHARC processors and contain 2-4 Mbits of internal memory, each on a 3.1 by 4.5 inch form factor. Each SHARC processor can simultaneously sustain 120 Mflop/s core computation and 240 MB/s of I/O via its DMA engines. The SHARC processors are interconnected via link port, shared memory or serial port I/O. This interconnection establishes the system as a mesh or network of nodes resulting in high-speed data compression of the raw images.

B. Transform-Based Compression Implementation Strategy

The BLAST algorithm has been implemented at HBOI using a single SHARC processor. Programming support is provided within a software environment called APEX (Advanced Parallel Executive) -- supporting both high and low level programming modes. A distributed kernel is provided that allows programming a large network without concern for communications at the hardware level. APEX provides a complete set of debugging tools, C compiler, assembler, runtime library, librarian, linker, and simulator. An emulator is also available for real-time hardware debugging.

Initial execution benchmarks for a 640 x 480 x 8 bit image compressed using BLAST are in Table 2. These benchmarks were achieved using non-optimized C language code executed on a single SHARC processor.

The total time shown in Table 2 can be improved through optimization of the DSP code and by partitioning the algorithm for operation using more than one SHARC processor. For example, a factor of 4 or more is possible using the 4 parallel processors currently in the system. Additionally, optimizing the inner loops of the copy, convolve, round and average functions are anticipated to increase throughput by an additional factor of 3-4.

PROGRAM MODULE	FUNCTION	TIME
CopyImg	Read image and pack 4 pixels into 32-bit word	3.7 s
PIConvolve	Convolve w/sharpening template and pack 4 pixels/word	6.1 s
PIRound	Round scale and normalize to full gray range	1.5 s
AverageBlocks	Compile block averages for KxL blocks	0.9 s
GrayscaleQZN	Quantize gray scale	0.1 s
WritePackedBinary	Write a packed binary file	0.1 s
TOTAL		12.4 s

Table 2. Initial execution benchmarks for non-optimized BLAST with single SHARC processor.

C. Host Computer Program

The host computer program is a multi-threaded application developed using LabVIEW 5.0. Optimized C/C++ functions have been added using Microsoft Visual C/C++. Tasks performed by this program include: configure frame grabber board; initiate image acquisition from frame grabber board; read acquired raw image; send raw image to DSP engine; receive compressed image from DSP engine; and send compressed image to acoustic modem. Although the SBC host running under NT provides a more flexible and robust development environment, we could further optimize the system by eliminating the host computer and program. In this configuration, the DSP engine would operate in stand-alone mode with direct interfaces to the data acquisition (or camera)

and acoustic modem. This would also allow a lower power diskless (ROM based) configuration which could be further optimized for specific missions or AUV environments.

III. PERFORMANCE MEASURES

The BLAST initial results demonstrate high speed data compression of 640 x 480 x 8 bit non-acoustic undersea images (shown in Figures 4a and 4b) and for live images acquired in a laboratory setting. There are several performance metrics we have used in evaluating the BLAST algorithms. These include compression/decompression timing and MSE comparisons vs. CR using standardized test images. A more comprehensive treatment of image degradation measures has also been developed for various compression methods [6]. Initial tests using an enhanced elliptical BLAST algorithm (EBLAST or BLAST-II) and randomized sub-sampling processing suggest improved visual acuity and reduced MSE for large compression ratios (> 200:1). Figure 4c is an example of EBLAST (CR = 240:1) with additional post decompression processing. The high computational horsepower available with the multiprocessor SHARC will permit this more sophisticated processing while maintaining acceptably high throughput.



Figure 4a. Original Coral image (480 x 640 x 256 gray levels).



Figure 4b. Compressed image using BLAST
(CR = 128:1).

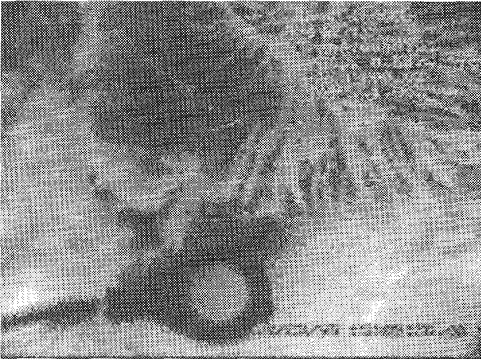


Figure 4c. Compressed image using EBLAST
(CR = 240:1).

IV. FUTURE PLANS AND DEVELOPMENTS

A DSP hardware implementation of a transform-based compression algorithm for AUV telemetry has been described and demonstrated in the laboratory. The system is scheduled for sea trials in November.

Several hardware and software/algorithm enhancements are planned. The simplest change is to optimize coding of the pack, unpack and convolve functions using low-level, in-line assembly code. Once the code is optimized, the algorithm can be partitioned across the image space to utilize multiple parallel DSP processors. This alone would result in a 2-3 time speedup. If further speed is desired, the host computer can be removed and captured images can be transferred

directly to the SHARC processors using PCI master/slave capability. The algorithm is also being refined to increase CR beyond 300:1 and to decrease MSE. (With the current image size, CR of 300:1 results in ~ 1 KB compressed file size.) Furthermore, the use of an information theoretic approach for hardware performance optimization is being explored as are advantages of architecture for adaptable reconfigurable and evolvable implementations. These enhancements would afford increased speed and efficiency required for specialized applications.

ACKNOWLEDGMENTS

This work was conducted under ONR Grant N0001496-1-5019 under the auspices of Dr. Thomas Curtin. This is Harbor Branch contribution number 1244.

REFERENCES

- [1] Schmalz, M.S., G. X. Ritter, and F. M. Caimi, "Performance Evaluation of Data Compression Transforms for Underwater Imaging and Object Recognition," *IEEE Oceans '97 Conference*, 2:1075-1081, 1997.
- [2] Schmalz, M.S., G.X. Ritter, and F.M. Caimi, "Data Compression Techniques for Underwater Imagery", *IEEE Oceans '96 Conference*, 2:929-936, 1996.
- [3] PC-Based Vision Solutions, National Instruments Corporation, part no. 350323B-01, April 1997 Edition.
- [4] Freitag, L., M. Johnson, J. Preisig, "Acoustic Communications for UUVs," *Sea Technology*, pp. 65-71, June 1998.
- [5] Neel, A., L.R. LeBlanc, J.C. Park, S.M. Smith, "Peer-to-Peer Communication Protocol," *Sea Technology*, pp. 10-15, May 1998.
- [6] Caimi, F.M., M.S. Schmalz, G.X. Ritter, "Image Quality Measures for Performance Evaluation of Compression Transforms," *Proc SPIE*, San Diego, 1998.