A Progress Report

Grant No. N00014-94-1--0841
June 20, 1994 - February 14, 1995

SOFTWARE FOR ADVANCED VISION SYSTEMS

Submitted to:

Dr. Thomas M. McKenna
Code 342CN
Office of Naval Research
800 N. Quincy Street
Arlington, VA 22217-5660

Submitted by:

Thomas J. Olson
Assistant Professor

SEAS Report No. UVA/525488/CS96/101
July 1995

DEPARTMENT OF COMPUTER SCIENCE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
July 1995

3. REPORT TYPE AND DATES COVERED
Progress Report 6/20/94 - 2/14/95

4. TITLE AND SUBTITLE
Software for Advanced Vision Systems

5. FUNDING NUMBERS
N00014-94-1-0841

6. AUTHOR(S)
T. J. Olson

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
University of Virginia
School of Engineering and Applied Science
Department of Computer Science
Thornton Hall
Charlottesville, VA 22903-2442

8. PERFORMING ORGANIZATION REPORT NUMBER
UVA/525488/CS95/101

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Office of Naval Research
800 North Quincy Street
Arlington, VA 22217-5660

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Unlimited

12b. DISTRIBUTION CODE
DISTRIBUTION STATEMENT A
Approved for public release
Distribution Unlimited

13. ABSTRACT *(Maximum 200 words)*

See report.

14. SUBJECT TERMS

15. NUMBER OF PAGES
2

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	Unlimited

NSN 7540-01-280-5500

Standard Form 298 (890104 Draft)
Prescribed by ANSI Std. 239-18
298-01

Software for Advanced Vision Systems

Progress Report - 6/15/95

Thomas J. Olson and Worthy N. Martin

The goal of the ARPA Advanced Vision Systems program (AVIS) is to develop hardware accelerators that will provide dramatic improvements in performance for machine vision, image processing, automatic target recognition, and other applications with similar processing needs. During phase I of the program our group conducted a study of software issues raised by the AVIS program. The goal of our current effort is to develop an abstract lower level interface for real-time computer vision accelerators. This interface will allow applications or development environments to control, sequence and synchronize with computations running on multiple accelerators without reference to vendor- or device-specific functionality.

Activities to date:

We delivered the final Phase I report on April 15th. One of the lessons of that effort was the importance of supporting the Khoros image processing and visualization environment. During the past three months we have been working to understand the Khoros process model and identify the options for supporting it on non-standard hardware. We plan to use this information to constrain the design of the abstract low-level API.

Khoros 2.0 is a complete rewrite of the Khoros system. It is written in C, but uses function tables to implement essential object-oriented language features such as dynamic inheritance. Any approach to supporting Khoros on AVIS architectures will likely make use of these features for the sake of maintainability and compatibility with other Khoros code. There appear to be two basic strategies for accessing special hardware via Khoros. One is to create a subclass of the Khoros *node* type. Nodes in Khoros encapsulate the computational behavior of a dataflow graph operator. It would not be difficult to create a node that spawns child processes on an accelerator to do the bulk of its computation. The problem with this approach is that it would leave gross scheduling decisions under the control of the Khoros executive.

A second approach to using special hardware via Khoros is to override the behavior of the Khoros Workspace object. It should be possible to create workspaces that support unique sets of nodes and schedule them according to unique protocols (e.g. static dependency graph scheduling). The advantage of this approach is that it should make it possible to use Khoros operators implemented as conventional workstation executables in combination with those that target an AVIS accelerator. This in turn would provide a smooth path for transitioning existing applications to the accelerator: Khoros operators could be recoded for the accelerator one at a time, providing incremental performance improvements.

The main technical problem that we see concerns the difficulty of using multiple accelerator-based workspaces in a single application. The low- and intermediate-level software for some AVIS accelerators may require a global view of the computation, e.g. in order to perform static scheduling and resource allocation. Khoros is currently conceived as a highly dynamic system in which decisions of that type are made at run time. We are currently designing a Khoros interface to our statically scheduled VEIL system, in order to ensure that the approach we choose can meet these requirements.

DISTRIBUTION LIST

1 - 3	Dr. Thomas M. McKenna Code 342CN Office of Naval Research 800 N. Quincy Street Arlington, VA 22217-5660
*	SEAS Postaward Research Administration
4 - 5	H. Earnhardt, Clark Hall
6	Director Naval Research Laboratory Washington, DC 20375 Attention: Code 2627
7 - 8	Defense Technical Information Center S47031 Building 5, Cameron Station Alexandria, VA 22314
9	Mr. Michael Karp Administrative Contracting Officer Office of Naval Research Resident Representative 101 Marietta Tower, Suite 2805 101 Marietta Street Atlanta, GA 20323-0008
10	SEAS Preaward Research Administration

JO#6500:ph