Parallel Processing for Computer Vision and Image Understanding

A. Choudhary
S. Ranka

CRPC-TR92253
1992

Center for Research on Parallel Computation
Rice University
P.O. Box 1892
Houston, TX 77251-1892

Appears in IEEE Computer, February 1992.

Parallel Processing for Computer Vision and Image Understanding

Alok Choudhary
Sanjay Ranka
Syracuse University

What is vision? From one point of view, "it is the process of recognizing objects of interest from images". The word "process" refers to some form of processing performed on the input data, which may be one or more images. The phrase "objects of interest" implies the existence of a context under which this processing must take place and the existence of a representation used in the processing. For example, if we were asked, "Is there a table in this room?", we would have some representation of a table in our mind and look for something similar to it in the room. In that process, we would ignore objects that did not look like a table. That is, we started with a model and looked for some instance of the model in the room. On the other hand, if we were asked, "Describe all the objects in this room", we would scan the room, form some representation of the objects, and match them to some objects that are part of our "knowledge base". However, we may or may not know what to expect in the room. Both problems can be considered "vision problems". A general vision problem, therefore, is considered to be an ill-posed problem from a computational perspective. Vision has fascinated researchers from various disciplines such as psychology, neural science, computer science and engineering, etc. for a long time and a good discussion on various aspects of vision can be found in [2].

1 The Role of Parallel Processing

Problems in computer vision are computationally intensive. Consider a sequence of images at medium resolution (512×512 pixels) and standard frame rate (30 frames per second) in color (3 bytes per pixel) [3]. This represents a rate of almost 24 million bytes of data per second. A simple feature extraction algorithm may require thousands of basic operations per pixel, and a typical vision system requires significantly more complex computations. Parallel computing is essential in solving such a problem.

Historically, the need to speed up image processing computations brought parallel processing into the computer vision domain. Most image processing algorithms are inherently parallel because they involve similar computations for all pixels in an image. This has inspired the development of array processors. For example, NASA Goddard Center's MPP [1] has a 128×128 processor mesh connected array specifically built for image processing. A mesh architecture is one in which each processor is connected to four neighboring processors; North, South, East and West. It is suitable for image processing, because its structure closely mimics that of a two-dimensional image, and it provides an efficient local communication structure.

The lack of efficient global communication capabilities in a mesh architecture, and the requirement for fast top-down and bottom-up image processing, led researchers to propose another architecture known as a "Pyramid". A Pyramid architecture normally consists of several levels of meshes in which the top level has one processor and each succeeding level has four times as many processors as its parent array. In addition, to the mesh interconnections within each level, each processor is also connected to its four children (except in the bottom layer), and to its parent (except the root). Therefore, a pyramid architecture maintains several levels of image representations simultaneously. Both mesh and pyramid architectures have contributed significantly to the understanding and development of new algorithms for image analysis and vision, and they have considerably influenced the subsequent designs of parallel architectures for vision.

Current Status

Parallel processing has taken tremendous strides in the last decade. It has enabled scientists to perform very large scientific computations, which were impractical a few years ago. But the immense computational challenge presented by vision is yet to be satisfied. In fact, compared to the impact of parallel processing in other areas, the impact of parallel processing in the vision domain has been minimal for several reasons.

A typical vision system requires integrating algorithms from diverse areas such as image processing, numerical analysis, graph theory, artificial intelligence and databases. There is no clear understanding and consensus on how to achieve this. Specific problems in integration can also be attributed to a lack of understanding of the vision process itself, even if the computations and parallelism of some individual components are well understood.

Currently, the dominant approach to characterizing vision computations is to classify the processing requirements into three levels; viz., low-level, intermediate-level and high-level. The most recent image understanding benchmark [3] embodies this characterization. Sidebar 1 presents a brief overview of each level of processing.

Begin Sidebar 1:

- **Low-Level Processing:** This level of processing is normally termed bottom-up processing [3]. Most image processing operations fall into this category. Input data for this level includes images or simple transformations of images. Computations in low level processing are regular, exhibit high spatial parallelism, and mainly involve numeric processing.

These computations are well suited both for SIMD (Single-Instruction-Multiple-Data) and MIMD (Multiple-Instructions -Multiple-Data) architectures, . Example algorithms include edge detection, filtering operations and connected component labeling.

- **Intermediate-Level Processing:** This category is called intermediate level because it conveniently serves to bridge the bottom-up (low-level) and top-down (high-level) processing [3]. Computations in this category manipulate symbolic and numeric data. Examples of symbolic data include edges, lines etc., commonly referred to as tokens. Processing on this level attempts to build a coalition of tokens to extract meaningful entities - for example, forming rectangles from lines. Computations are normally data dependent and irregular. They are suitable for medium to coarse grain parallelism in MIMD mode, although a subset of computations also can be performed efficiently on SIMD architectures.

- **High-Level Processing:** Tasks on this level of processing are normally top-down (or model directed). However, processing on this level is not as well defined as on the other two levels. The model of the "world" (the world represents a database of objects, their possible poses, and interrelationships in a context, e.g., a model of a car will have descriptions of wheels, doors etc., and constraints describing their relationships) drives the processing on this level. Furthermore, processing in this domain may require re-executing algorithms from the other two levels. Although parallelism on computations in this level is not well understood, it is believed that dynamically scheduling of computations is needed. The diversity and highly data dependent nature of computations makes this level of processing largely suitable for MIMD parallelism.

End Sidebar 1:

Recent efforts in architectural design and development have attempted to embed architectural components suitable for each level of processing into one integrated architecture (see the sidebar). It can be said, however, that compared to the progress in architectural advances for general-purpose parallel processing for other desciplines of sceinces, architectural advances for vision systems are in their infancy.

2 Future Directions

The federal government's High Performance Computing and Communication (HPCC) initiative recently identified problems known as "Grand Challenges", and artificial vision falls into that category [4] (see Figure 1). Solving these problems is expected to require raw computational power between 100 and 10,000 billions of operations per second. Besides the raw computational needs, there are other issues that must be addressed to make efficient use of parallel processing in solving vision problems. The following is a brief overview of some important issues.

Architectures: As discussed earlier, vision problems span a broad spectrum of computations, which are suitable for different types of architecture (e.g, special purpose, SIMD, MIMD, etc.). The computations in various stages of processing not only need to execute concurrently in a system, they also need to interact with each other. Hence, the future points towards some form of heterogeneity in an architecture for vision. The challenge for researchers is to capitalize on the advancements in multiprocessor technology, and incorporate them optimally in an architecture suitable for vision problems.

System Integration: A straightforward integration of different types of architectures into one organization may not be efficient for solving vision problems. The integration must incorporate appropriate control and communication structures suitable for providing the necessary

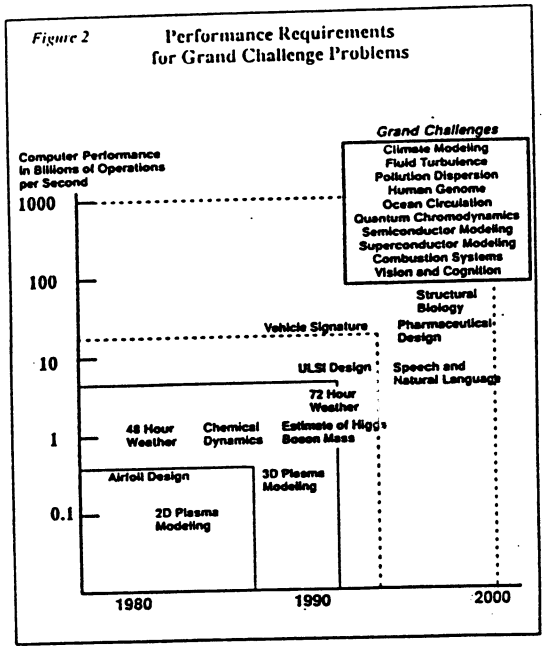

Figure 1: Performance Requirements for Grand Challenge Problems [4]

interaction between various parts, and it should reflect the peculiarities of vision systems. In simple terms, an integrated organization should not be visible to a user as a collection of heterogeneous components, but it should provide a unified view of a machine in which heterogeneity is transparent to the user.

Programming Models and Software Tools: Development of programming models and software tools has lagged far behind the progress in architectures. Vision processing requirements present a greater challenge because vision requires programming models, and subsequently, languages which are powerful and flexible enough to handle different types of processing, but at the same time, capable of hiding the details of a heterogeneous system.

Software tools are necessary to provide users a friendly platform on which they can develop applications. If users are burdened with the responsibilities of learning machine organizations intimately in order to use them, the expected gains from parallel processing will be limited.

Tools are also important for users to rapidly prototype and experiment with their algorithms. This is critical in order to develop and test new techniques to solve problems in the vision domain.

Real-Time Vision: Examples of real-time vision applications include robotics and autonomous land vehicles. Such applications not only demand high performance, but they require predictable performance, and a capability to interact with the environment (e.g., active vision) through efficient interactions with sensors. They also place physical constraints such as size, power consumption, cost and robustness on a parallel system. Therefore, special-purpose parallel computing may yet have a role in such systems.

3 Acknowledgments

The guest editors would like to thank Dr. Jon T. Butler, editor-in-chief of *Computer* for his invaluable help and guidance in all aspects of this issue including gathering reviews, interpreting their results, selecting papers and shaping the theme. We would also like to thank Dr. Bruce Shriver, the former EIC, for his encouragement and subsequent approval for this special issue. We acknowledge the support of the Center for Applications and Software Engineering, a New York State Technology Center at Syracuse University, Department of Electrical and Computer Engineering and School of Computer Science. We wish to thank all the reviewers for their timely and invaluable reviews, and all the authors for their contributions and discussions with us. We are grateful to Professor Azriel Rosenfeld for sharing his views on the theme of the issue and for many enlightening discussions. We would like to acknowledge support from Texas Instruments and IBM. Finally, we thank Grace Lanni and Carla Shaw for their professional handling of manuscripts and reviews.

This work was partially supported by DARPA under contract # DABT63-91-C-0028. The content of the information does not necessarily reflect the position or the policy of the Government and no official endorsement should be inferred.

References

[1] K. E. Batcher, "Design of a Massively Parallel Processor," *IEEE Trans. Computers,* pp. 836-840, 1980.

[2] D. Marr, "Vision," W. H. Freeman and Co., San Francisco, CA, 1982.

[3] C. Weems, E. Riseman, A. Hanson and A. Rosenfeld, "The DARPA Image Understanding Benchmark for Parallel Computers," *Journal of Parallel and Distributed Computing,* Academic Press, pp. 1-24, Jan. 1991.

[4] "Grand Challenges: High Performance Computing and Communications," A Report by the Committee on Physical, Mathematical, and Engineering Sciences, Office of Science and Technology Policy, 1991.

Sidebar:

Further Reading

In total, 25 submissions were received for this special issue of *Computer*. The eight article selected represent a high quality, but unfortunately, many other interesting papers could not be included. For more information consult some of the sources listed below for additional material.

Journals: IEEE Computer, IEEE Transactions on Computers, IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions on Parallel and Distributed Processing, Journal of Parallel and Distributed Computing (Academic Press), Journal of Supercomputing (Kluwer Academic), International Journal on Computer Vision (Kluwer Academic), Computer Vision Graphics and Image Processing (Academic Press).

Conferences: Image Understanding Workshop (Morgan Kaufmann), Conference on Computer Vision and Pattern Recognition, International Conference on Pattern Recognition, International Conference on Parallel Processing (Penn State Press), IEEE Computer Society Workshop on Computer Architecture for Pattern Analysis and Machine Intelligence.

Books:

1. A. N. Choudhary and J. H. Patel, "Parallel Architectures and Parallel Algorithms for Integrated Vision Systems," *Kluwer Academic Publishers,* 1990.
2. V. Kumar, P. S. Gopalakrishnan and L. N. Kanal, "Parallel Algorithms for Machine Vision and Intelligence," Springer Verlag, 1990.
3. V. K. Prasanna Kumar, " Parallel Architectures and Algorithms for Image Understanding," Academic Press, 1991.
4. S. Ranka and S. Sahni, Hypercube Algorithms for Image Processing and Pattern Recognition, Springer, Berlin, 1990.
5. L. Uhr, ed., Parallel Computer Vision, Academic Press, Boston, MA, 1987.

BIOGRAPHIES

Alok Choudhary has been on the faculty of the Department of Electrical and Computer Engineering at Syracuse University, Syracuse, NY, since 1989. His research interests include parallel computer architectures, software development environments for parallel computers and computer vision.

Choudhary received his B.E. degree in Electrical and Electronics from Birla Institute of Technology and Science, Pilani, India. He obtained M.S. degree from University of Massachusetts, Amherst, and his PhD degree from University of Illinois, Urbana-Champaign, both in Electrical and Computer Engineering. He was a visiting scientist at IBM, T.J. Watson Research Center during the summers of 1987 and 1988. He worked as a system analyst and designer from 1982-1984. He is a member of the IEEE Computer Society and the Association for Computing Machinery.

Sanjay Ranka completed his B. Tech in Computer Science and Engineering from Indian Institute of Technology, Kanpur in May 1985 and Ph. D. in Computer and Information Science from the University of Minnesota, Minneapolis in August 1988.

Since August 1988, he is an Assistant Professor in the School of Computer Science at Syracuse University. He was a visiting scientist at IBM T. J. Watson Research Center during summer of 1991. His main areas of interest are parallel and distributed computing, algorithms, software development environments for parallel computers, and neural networks. He is a member of the IEEE Computer Society.