

New Architectures for Vision

Kendall Preston, Jr.

Kensal Consulting, 5055 East Broadway #C206  
Tucson, AZ 85711 USA

Abstract

Over the years computer architectures for vision applications have evolved into the five basic types described in this paper. The simplest use a single PE (processing element), a primary store holding one or more images, and a secondary store that acts as a buffer between the primary store and the PE. However, more and more vision systems are now trending to type five, i.e., massive arrays of PEs that have either one PE dedicated to each column of the image and accessing multiple rows simultaneously or, in many cases, one PE dedicated to each pixel in 2D imagery or even to each voxel in 3D imagery. Lastly, these architectures, thanks to advances in VLSI, are taking less and less space and using less and less power. We find today on a single chip systems that were rack-mounted monsters 20 years ago.

Introduction

As pointed out by the author in a recent article (Preston, 1992), the cellular automaton or massively parallel processor architecture leads the way in vision analysis both in processing speed and cost effectiveness. More and more machines, based on the cellular automaton architecture, have entered the market during the 1990s. In the latest Abingdon Cross comparison (Figure 1) the GAPP II, III, and IV of Martin Marietta and the MP-1208 and MP-1216 of Maspar, the Zephyr-8 of Wavetracer, the Viper of Amber Engineering as well as the AIS5000 linear array automaton of Applied Intelligent Systems lead the field. Also note the performance of earlier cellular automata from the 1980s such as the MPP (Massively Parallel Processor) of Goodyear Aerospace and the CM (Connection Machine) of Thinking Machines Company.

Thus this paper on new architectures for vision will concentrate first on these systems and then on other architectures of interest that are not cellular automata but have other advantages such as low cost (cellular

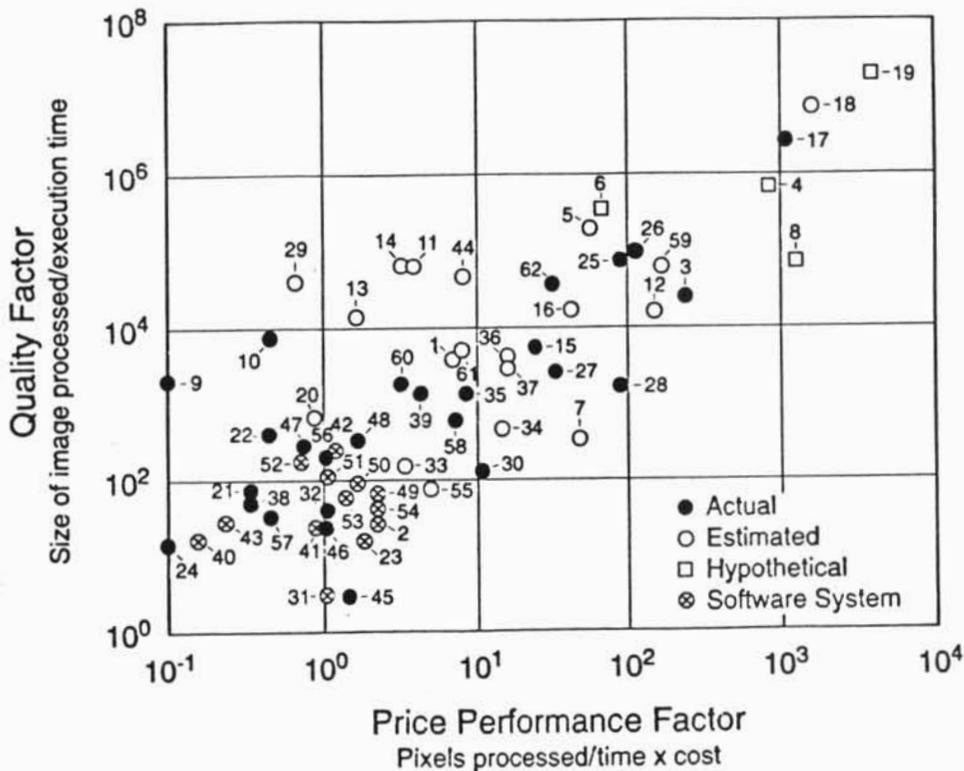


Fig. 1 - Latest Abingdon Cross results (with permission Advanced Imaging, copyright Sept. 1992).

automata are still very expensive) and compactness (cellular automata are still very large).

## Historical Perspective

Lest we imagine that cellular automata are a new idea - many newcomers are claiming that they have "invented" these massively parallel architectures - let us remember our history. First to imagine that arrays of computers could be a useful architecture was von Neumann in the 1940s. After being introduced to ENIAC (Electronic Numerical Integrator and Computer), the first all-electronic computer built in the USA (Goldstine, 1972), he formed a task force that included Howard Aiken (Harvard) and Norbert Wiener (MIT) to study electronic computers and their applications. Because von Neumann's interests also encompassed neural networks in the brain (McCulloch and Pitts, 1943), von Neumann included in the charter of the task force studies on the "communication and control aspects of the nervous system." Inspired by the ideas of others on general automata (Post, 1936, and Turing, 1936) von Neumann not only designed the first stored-program computer (EDVAC) but also began work on the design of systems that would be self-reproductive.

Conversations with colleagues at the USA Los Alamos National Laboratories (Ulam, 1962) convinced him to pursue research on arrays of "processing elements." His ideas on this subject were put into writing in the early 1950s and then were introduced in a series of lectures at Princeton University summarized in an article in the Scientific American (Kemeny, 1955). Although von Neumann's premature death in 1957 prevented completion of these formulative concepts, the idea of the cellular array computer or "cellular automaton" had been born.

## Cellular Logic

Research in the application of cellular automata to vision using what became known as "cellular logic transforms" then began at MIT Lincoln Laboratory (Dinneen, 1955), Bell Telephone Laboratories (Unger, 1958), and the Perkin-Elmer Corp. (Golay, 1969, and Preston, 1961). The world's first dedicated cellular logic machine (CELLSCAN) was built at Perkin-Elmer so as to reduce to practice Golay's concept of a hardware emulator of a cellular array. At the same time, general-purpose computers were used to emulate cellular logic arrays at MIT (Gardner, 1971, and Banks, 1970) where significant simplifications of von Neumann's processing elements were devised. Still simpler arrays were simulated at Los Alamos (Schrandt and Ulam, 1960) using two-state computing elements. Additional studies were carried out at the University of Pennsylvania and Fort Monmouth (Yamada and Amaroso, 1971), at the Polytechnic Institute of Brooklyn (Smith, 1971) and, in Japan (Tojo, 1967, 1968, and 1970). A major conference on cellular automata was held in the mid-1970s sponsored by the IEEE (1975) with a second meeting in the following decade under the auspices of the USA Department of Energy (Farmer, et. al., 1984).

## Large Scale Integration

Most of the above were academic studies, except CELLSCAN, due to the fact that reductions to practice in hardware would have been prohibitively expensive. With the advent of large scale integration (LSI) in the 1970s, full-array cellular automata were proposed (Slotnick et. al., 1962, and McCormick, 1963) and

Figure 1 Legend

- 1 1024XM (Megavision)
- 2 AIM (Delft University on Atari 1040ST)
- 3 AIS 5000 (Applied Intelligent Systems)
- 4 ASP 400 (Brunel University)
- 5 CAAPP64 (University of Massachusetts)
- 6 CAAPP512 (University of Massachusetts)
- 7 CAM-6 (System Concepts)
- 8 CENTIPEDE (Applied Intelligent Systems)
- 9 CM-2 (Thinking Machines - using C\*)
- 10 CM-2 (Thinking Machines - using PARRIS)
- 11 CM-2 (Thinking Machines - expected)
- 12 CYTO-HSS (Environmental Research Inst. of Mich.)
- 13 DAP 32x32 (International Computers Ltd.)
- 14 DAP 64x64 (International Computers Ltd.)
- 15 DAP 510 (Active Memory Technology)
- 16 DAP 610 (Active Memory Technology)
- 17 GAPP II (Martin Marietta)
- 18 GAPP III (Martin Marietta)
- 19 GAPP IV (Martin Marietta)
- 20 GE/WARP (General Electric)
- 21 IP8500 (ETH Zurich)
- 22 IP9200 (Perceptics)
- 23 JAVA (Jandel Scientific on IBM AT/386)
- 24 Magiscan-2 (Joyce-Loebl)
- 25 MP-1208 (MasPar)
- 26 MP-1216 (MasPar)
- 27 MaxVideo (DataCube)
- 28 MP150 (Noesis Vision)
- 29 MPP (NASA Goddard)
- 30 MVP/AT (Matrox)
- 31 PhotoSynthesis (Escape Sequence on Amiga)
- 32 PIP 4000 (ADS Company Ltd.)
- 33 PIP 4500 (ADS Company Ltd.)
- 34 PIP 9000 (ADS Company Ltd.)
- 35 PIXAR/1-ChaP (Vicom-Pixar)
- 36 PIXAR/3-ChaP (Vicom-Pixar)
- 37 Pixel Machine (American Telephone & Telegraph)
- 38 PSICOM 327 (Perceptive Systems)
- 39 Scope-20 (Symbolics)
- 40 Semper 6 (Symbolics on DEC MicroVax 11)
- 41 Semper 6 (Symbolics on DEC AT/386-20)
- 42 Semper 6 (Symbolics on Silicon Graphics 4D20)
- 43 Semper 6 (Symbolics on Sun 3/150)
- 44 SPDS (Amber Engineering)
- 45 T800 (Scottish Reg. Transputer Support Centre)
- 46 T800 Array (Scottish Reg. Transput. Support Centre)
- 47 TAS-Plus (Leitz GmbH)
- 48 TERAGON (Teragon)
- 49 TCL Image (Delft University on Macintosh II)
- 50 TCL Image (Delft University on Sun 3/60)
- 51 TCL Image (Delft University on Sun 4/110)
- 52 TCL Image (Delft University on Sun 4/280)
- 53 TIM (Delft University on Compaq AT/386-16)
- 54 TIM (Delft University on Tulip AT/286-10)
- 55 TMS340 (Applied Imaging MAD-pack)
- 56 TOSPIX II (Toshiba)
- 57 TRAPIX 5500 (Recognition Concepts Inc.)
- 58 VICOM VME-II (Vicom)
- 59 VIPER (Amber Engineering)
- 60 VITec-1 (Visual Information Technologies)
- 61 VITec-2 (Visual Information Technologies)
- 62 Zephyr-8 (Wavetracer)

finally built by the Burroughs Corp. in the form in the ILLIAC IV (Slotnick, 1971). The ILLIAC IV, having 256 processors and costing tens of millions of dollars, was finally installed and used in the 1970s at the USA National Aeronautics and Space Administration Ames Laboratory in California. At more or less the same time, researchers at University College London began work on a series of full-array cellular automata called CLIP (Cellular Logic Image Processor). The third in this series, CLIP3, became operational in 1973 (Duff and Watson, 1975) and CLIP4 in 1980 (Fountain, 1983). CLIP4 was a 96x96 cellular automaton composed of some three million transistors integrated at three thousand per chip. This machine, as well as the 64x64 DAP (Distributed Array Processor) of International Computers, Ltd. (Hunt, 1981) and the 128x128 MPP (Massively Parallel Processors) of Goodyear Aerospace Corp. (Batcher, 1980) truly represented the fruition of von Neumann's visionary work in the 1950s.

Even with the introduction of VLSI (Very Large Scale Integration) in the 1980s, costs were still high. The development cost of the MPP was several million dollars. However, the next series of cellular automata to be introduced, namely, the CM-1 and CM-2 of Thinking Machines Corp., reduced this price, at least for a small machine (128x64) to of the order of one million dollars. Further price reductions in the 1990s by Maspar and Wavetracer have resulted in 8,192-processor machines for only a few hundred thousand dollars.

### Fundamental Architectures

Before proceeding with a detailed description of the architecture of cellular array computers, the five major architectural types should be described. Each architecture comprises of one (or more) primary image store(s) that contain the full image. The primary image store(s) connects to one or more processing elements (PEs) via one (or more) secondary store(s) that hold some portion of the full image. Transfer paths from these stores to the PEs may be binary, moving one bit at a time, or multilevel, moving many bits in parallel. CELLSCAN and similar machines (Gray, 1972, and Kruse, 1973) used a single high-speed PE that operated on data contained in a 3x3 subarray that was part of the secondary store. The primary memory for CELLSCAN was magnetic tape and, in other subarray machines was furnished by the main memory of a host computer. This relatively simple architecture is shown in Figure 2a.

As time progressed another architecture emerged (Figure 2b) that used multiple primary stores with a secondary store that furnished data to many PEs (Graham and Norgren, 1980). These PEs operated upon many data values simultaneously, all of which were extracted from the secondary store in parallel. Currently this is a popular architecture and many such machines employing this architecture are deployed worldwide. Another architecture, called the pipeline architecture (Figure 2c), has been developed by the Environmental Research Institute of Michigan (Sternberg, 1981) and elsewhere (Nawrath and Serra, 1974). Here, many separate PEs are pipelined as a string of identical CELLSCAN-like stages with the output of one being coupled directly to the input of the next. Each machine has associated with it the full complement of CELLSCAN-like secondary storage registers.

An alternative architecture is that given in Figure 2d where the secondary stores are multiplexed to several processing elements in parallel. Finally, there is the full-array machine, e. g., ILLIAC, CLIP, DAP, MPP, CM, GAPP, MP, and Zephyr, that is shown in Figure 2e. Even in this case there exists an off-line primary store.

### Specific Examples

Since it is impossible, even in a long paper, to describe all of the cellular automaton architectures that have been or are being used in vision, this section presents a few specific examples of machines representing the architectures shown in Figure 2.

#### CELLSCAN

CELLSCAN, built in 1961, is shown in Figure 3. It is an example of the architecture shown in Figure 2a. It was a true "turing" machine that both read from and wrote on an endless magnetic tape at 3,720 picture elements per second. This tape formed the primary store. CELLSCAN was self-contained (using no host

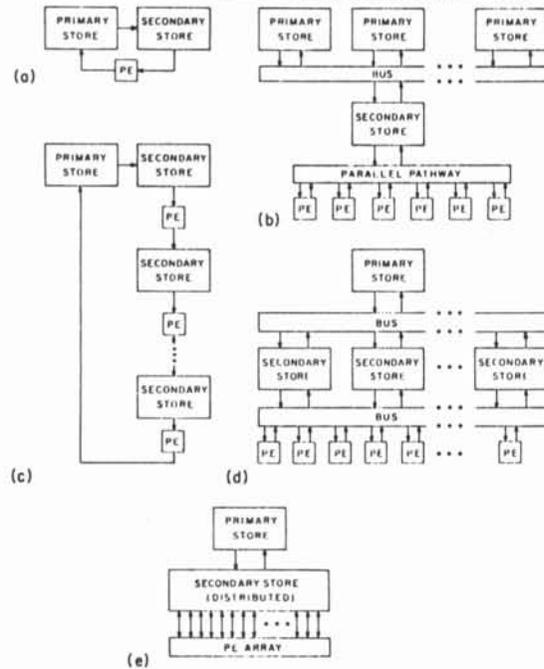


Fig. 2 - The five basic types of vision architectures.

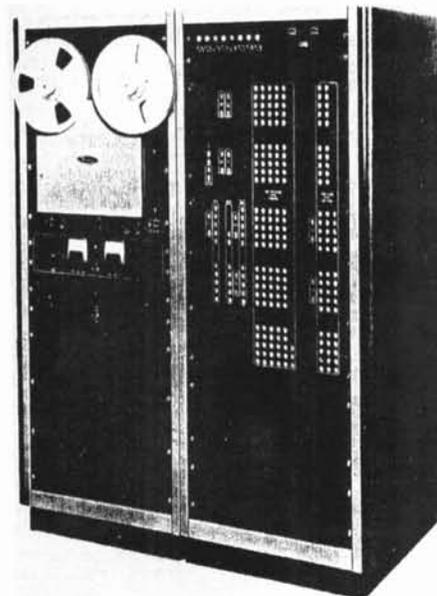


Fig. 3 - CELLSCAN

computer) as designed at Perkin-Elmer (Preston, 1961). Its secondary store consisted of two 60-bit shift registers holding the incoming picture element values and another 60-bit shift register holding the values of the picture elements after processing, all implemented in semiconductor memory (Figure 4). As can be seen in Figure 3, CELLSCAN occupied two entire racks of equipment. As indicative of the dramatic reduction in size from rack-mounted vision systems to single VLSI chips, the Applied Physics Laboratories of Johns Hopkins University reduced the basic CELLSCAN processing element to a single chip in the 1980s.

#### AIS5000

A series of linear-array cellular automata have been commercialized by AISI (Applied Imaging Systems, Inc.) of Ann Arbor, MI. These are typical of the architecture shown in Figure 2b. A multiplicity of primary stores receiving images from a multiplicity of television cameras are connected to a full-frame secondary store that feeds all columns in parallel, row by row, to array of up to 1024 PEs. The PE of the system is shown in Figure 5. Each PE receives inputs from three rows of the image in parallel and executes binary neighborhood functions by means of a 16-position LUT (Look-up Table). With a clock rate of 10MHz an entire 1024x1024 may be transformed in 100 microseconds! This makes the AIS5000 three hundred times faster than real-time where "real-time" means one transform each 30 milliseconds (the standard frame time of a television camera). In other words, the AIS5000 can perform a 300-instruction program at television frame rates making it possible to do enormously complex vision tasks at what are called "video rates." This explains its remarkable performance on the Abingdon Cross with extraordinarily high cost-effectiveness (sell price in the vicinity of 50 thousand dollars).

#### The Cytocomputer

The Environmental Research Institute of Michigan Cytocomputer (Sternberg, 1981) is an example of the vision architecture shown in Figure 2c. It connects a series of CELLSCAN-like PEs in series. Each PE contains secondary storage in the form of two fixed-length registers, each with 509 stages plus the 3x3 subarray register for computing neighborhood transforms. The nine elements of the subarray register deliver a nine-bit word to a 512-position LUT. In such a pipeline structure (Figure 6) each LUT represents a different step or instruction in the processing cycle and therefore, must be loaded separately.

The host computer provides primary image storage and delivers the image data as a stream of multibit words from which the desired bits are gated to the LUTs of a given PE. This architecture is excellent when executing a known algorithm repetitively. In this case the contents of the LUTs can remain unchanged. In an N-position Cytocomputer, the processing speed is increased by a factor of N in comparison with a single PE system, assuming, of course, that all N stages are required for the algorithm being performed.

#### The PHP

The next advance was to the architecture of Figure 2d where multiple primary stores would transfer their data to a multiplicity of secondary stores whose contents was operated upon by a multiplicity of PEs. One example of this architecture is the Carnegie-Mellon PHP (Herron, et al., 1982). Architecture of the PHP is shown in Figure

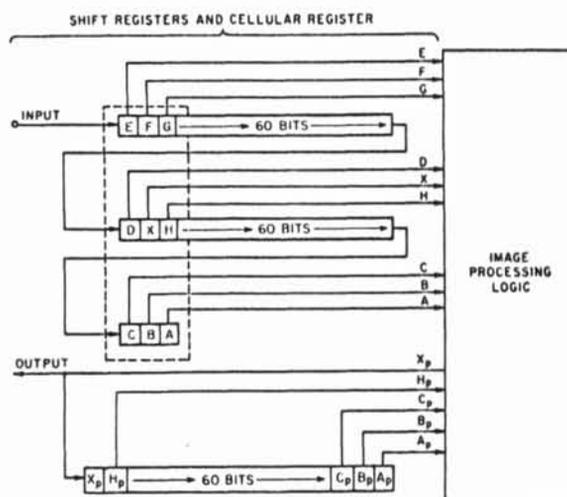


Fig. 4 - The CELLSCAN processing element.

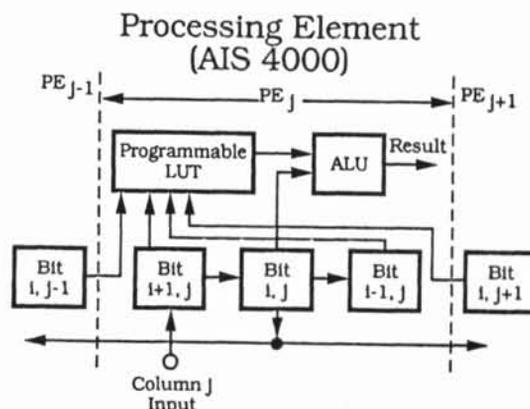


Fig. 5 - The processing element of the AIS4000.

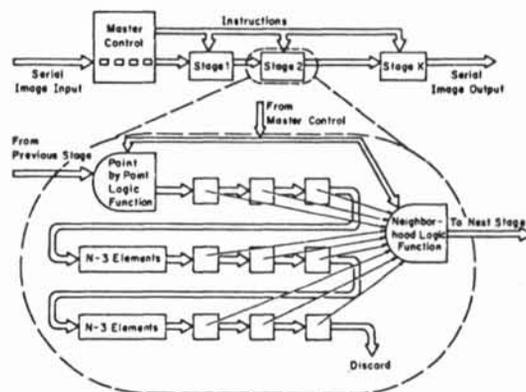


Fig. 6 - Block diagram of the Cytocomputer.

7. There are 16 identical processing elements (Table 1, Table 2, ..., Table 16) each configured as a LUT and all operating in parallel. Data was delivered to them from three secondary image stores that receive identical image data from the primary image store. Unlike CELLSCAN, these stores are not shift-registers but instead employ RAMs so that image information stored in them may be extracted in parallel from three separate lines of the

image. The line length is determined by three offset addresses (set externally by the host computer) that point to the desired data in the three image lines simultaneously. Since 16 processing elements must be serviced in parallel with data from three lines in the image, 54 bits of image data are delivered simultaneously from a set of buffers associated with the PEs. These buffers present 54 inputs to a distribution matrix whose 144 outputs furnish the 16 LUTs with 16 nine-bit addresses. The outputs of the PEs are gated within the PHP to the host computer via an output mask.

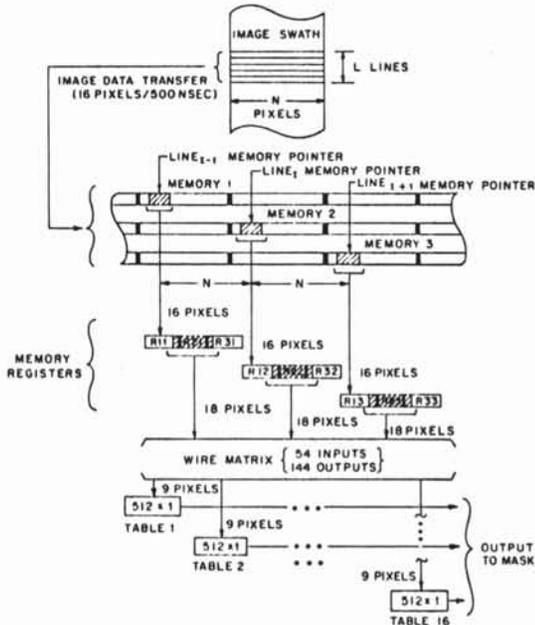


Fig. 7 - Block diagram of the Preston-Herron Processor.

### GAPP

The GAPP (Geometric Arithmetic Parallel Processor) of Martin Marietta, Orlando, FL, leads all other vision architectures in terms of both speed and cost effectiveness (Figure 1). As can be seen, the QF (Quality Factor) is 2.5 million, since the execution time of the Abingdon Cross benchmark is 42.4 microseconds for a 104x104 image. At a production price estimated at 220 thousand dollars, the PPF (Price Performance Factor) is 1160. It should also be pointed out that GAPP II is in actuality a 480x132 array of PEs making it possible to execute a 480x480 Abingdon Cross using five overlapping 480x120 windows. In this case, although the QF would, as expected, remain unchanged, the PPF jumps to nearly five thousand.

Martin Marietta is now constructing the first GAPP III systems that have double the clock rate of GAPP II (20MHz). In this case the QF will increase to between four and five million while the PPF would climb to over 13 thousand. This puts the family of GAPP cellular automata significantly ahead of all other vision architectures.

The detailed architecture of GAPP is therefore worthy of careful attention. In contrast to other cellular automata that employ thousands of bits of off-chip RAM per processor, GAPP survives handily with only 128 bits, all of which are on-chip. This avoids slow off-chip memory access and, furthermore, permits the integration of from 72 to 128 processors per chip (versus only from

8 to 32 in other machines such as CM, MP, and Zephyr). The next generation (GAPP IV) is being designed with 192 processors per chip and will support, when necessary, off-chip memory access.

The GAPP PE is shown in Figure 8. This PE supports parallelism within parallelism. This is obtained by executing more than one event per instruction cycle. Each instruction is executed in one clock cycle but up to five simultaneous events may occur during this clock cycle. Examples would be writing the result of the previous clock cycle to on-chip memory, simultaneously loading two operands from two neighboring processors, setting an internal latch, and shifting a previously loaded bit plane by one row across the array. This latter capability permits image I/O to take place simultaneously with image processing. This and other features are what give the GAPP-series of machines a significant advantage over competing vision architectures.

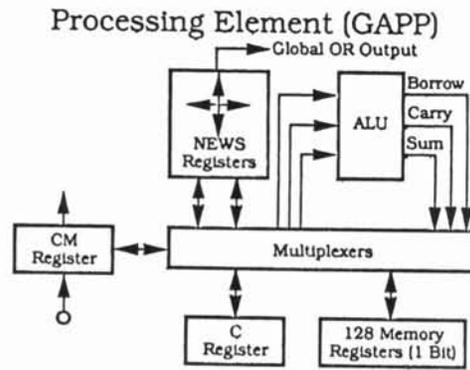


Fig. 8 - The GAPP processing element.

### Other Architectures

Numerous other vision architectures have been developed over the years and are reviewed in various handbooks (Young and Fu, 1986) in chapters on this subject (Preston, 1986). They include, but are not limited to, the BIP that is the image processor of the GRAFIX 1 system manufactured by Information International, Inc. (Gray, 1971), the PICAP (Picture Array Processor) of the University of Linkoping (Kruse et. al., 1980), the DIP (Delft Image Processor) of the University of Delft, and many, many others (see legend for Figure 1). Systems produced during the 1980s were rack-mounted. In the late 1980s and 1990s we see a trend to the "box" configuration as exemplified by PIP-series machines from ADS, Ltd. (Osaka, Japan) and the FIRE-PIP of Nippon Steel (Fuchinobe, Japan), and others. From the "box" configuration we now see a trend to vision systems that are "single-board" structures and some which have been reduced to a single VLSI chip. This section reviews some of these developments.

### IP8500

Probably the most famous of the "rack-mounted" systems is the IP8500 of DeAnza (Sunnyvale, CA) that was employed extensively during the 1980s worldwide. Its architecture is diagrammed in Figure 9. As can be seen there are four memory modules (512x512) that are connected through a multiplexor to both the display controller and the digital video processor. The performance of the system is determined by the contents of 105 control registers. The image memories are

# De Anza IP8500

(A/N graphics, cursor - not shown)

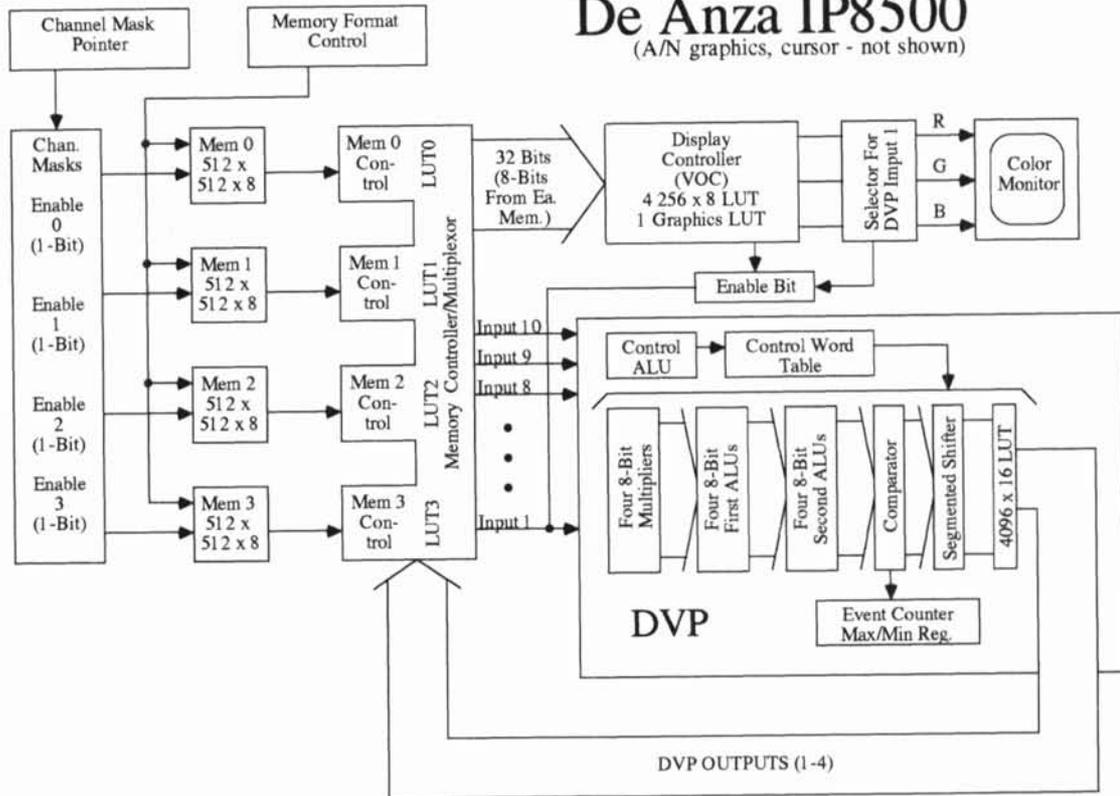


Fig. 9 - Block diagram of the DeAnza IP-8500.

individually enabled by the contents of the "channel mask registers" that are 32-bit registers. These permit the four memories to be configured as a single 1024x1024 or four separate 512x512 entities.

There is also a general memory controller containing four 16-bit registers and one 32-bit register used to set the starting address in memory. In addition each image memory has specific memory control registers with five being assigned to each memory module. Two of these provide the starting address for input to the digital video processor and also control zoom, wraparound, enabling of the display LUT, and controlling interlace for display purposes. Additional memory control registers contain masks that permit the digital video processor to store its output in particular bit planes of a particular memory. There are also corresponding masks for reading. Another register selects one of the four paths from the digital video processor and enables this path while controlling on-board LUTs. These LUTs are an important feature of the IP8500. Each memory board has four 256-position LUTs to which the LUT control register contains a starting address pointer as well as a table-select pointer.

There are ten inputs to the digital video processor that come from the multiplexor handling the output to the image memories. Eight of these inputs go pairwise to four 8-bit multipliers, followed in a pipeline arrangement by a bank of four 8-bit adders, followed by four additional 8-bit adders. The operation of these multipliers and adders are controlled via a control word LUT. The address of the control word LUT is determined by a separate control word arithmetic unit whose inputs are the two remaining inputs to the digital video processor. Thus the performance of the multipliers and adders may be modified in real time on a

pixel-by-pixel basis according to the inputs to the digital video processors that are connected to the control word arithmetic unit whose outputs, in turn, address the control word LUT. Furthermore, by combining the members of the bank of multipliers and the two banks of arithmetic units, it is possible to carry out both 16-bit and 32-bit computations.

As can be seen the IP8500 is an extraordinarily complex yet extraordinarily flexible vision system. Its architecture is probably the most complicated ever configured. This leads to programming complexity in that the assembly-level programmer must correctly enter the contents of all 105 control registers as well as correctly specify the contents of all LUTs. However, the popularity of this system attests to its usefulness in image processing, at least in the 1980s. Further, it has served as a model for many of the vision architectures of the 1990s.

## FIRE-PIP

Typical of modern image processing "boxes" is the FIRE-PIP of Nippon Steel (Fuchinobe, Japan) See Figure 10. The electronic technology of this machine is based on the parallel processing T800 chip from INMOS (Great Britain). From 8 to 20 of these chips are used for feature extraction and pattern recognition, interconnected by a fast (108 MB/s) image bus. Two additional T800 chips are employed in what is called the "Supervisor Preprocess" unit. Also interconnected over the bus are a 640x480 display system having 3MB of memory, digital I/O over SCSI, GPIB, and R5232 links, plus up to 256 MB of image memory (32 MB per board). The entire system is hosted by a NEC personal computer and SUN workstation.

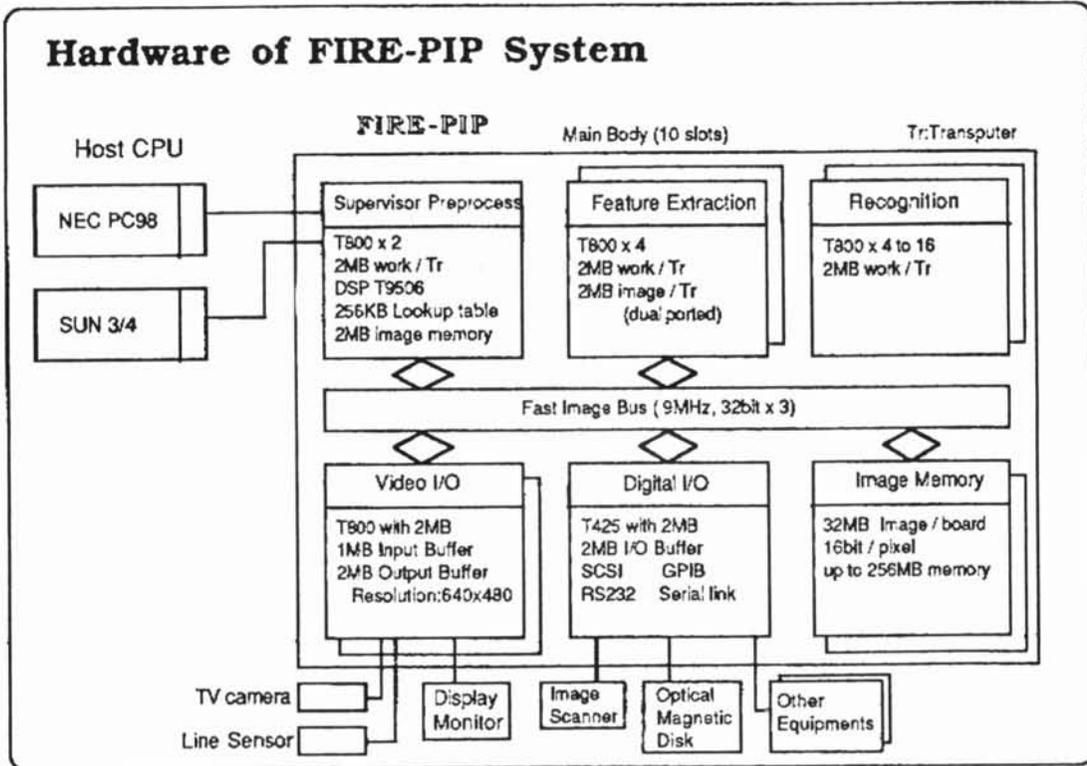


Fig. 10 - Block diagram of the Nippon Steel FIRE-PIP.

The CL-PX2070

Looking forward to 1993 when new single-chip vision systems will become available, let us take the CL-PX2070 as an example. The block diagram of this single-chip video processor is shown in Figure 11. As

with the IP8500, its performance is controlled by a multiplicity of registers (approximately 200). It consists of two video controllers that interface with television inputs. There is also a host interface unit, a video processor having full arithmetic and logic capabilities, its instruction sequencer, plus a processed video output unit

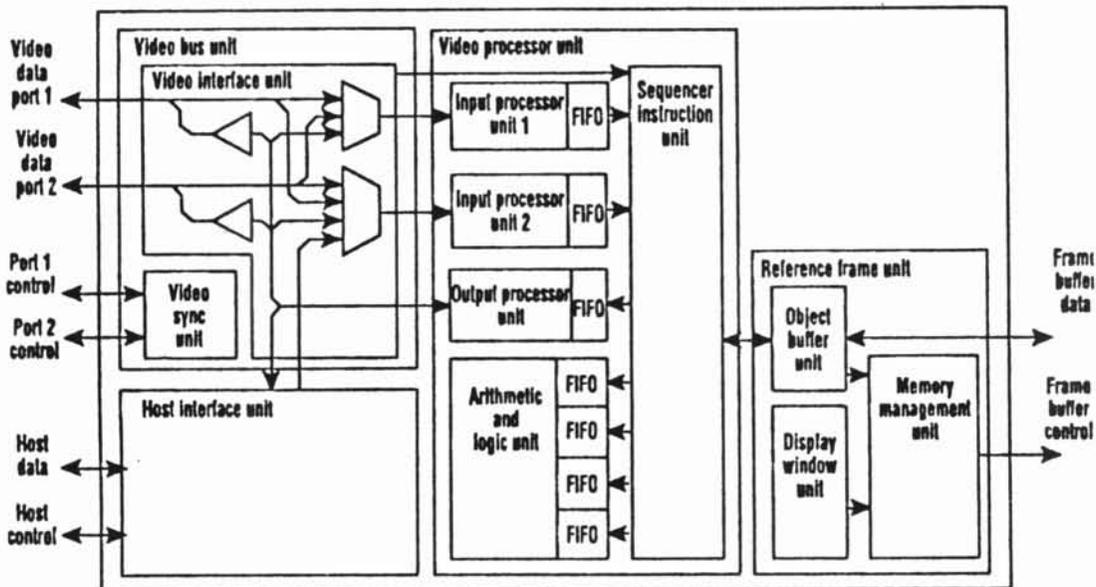


Fig. 11 - Block diagram of the CL-PX2070 - A single chip vision processor for video graphics (with permission Electronic Design, copyright Aug. 1992).

connecting to an external image frame buffer. The host interface permits connection to standard buses such as ISA (Industry Standard Architecture) or MCA (Micro Channel Architecture). The video I/O unit not only manages external television data sources but also may combine with these graphics data streams originating in the host. The two real-time video I/O ports are asynchronous and can be individually configured. Each has an on-chip sync generator that is programmable.

The video processor contains five separate blocks with associated FIFO (First In First Out) buffers, two input processors, an output processor, and a fully-configured arithmetic logic unit. These subsystems are controlled by instructions contained in the instruction sequencer. Amongst the many operations possible in this unit are format conversion, color scaling, independent horizontal and vertical scaling, and windowing. Individual pixels may be tagged and separately processed by the arithmetic unit in real time either separately or combining the two incoming streams of video data.

The instruction sequencer is essentially a special-purpose microcontroller that is capable of executing instructions at faster than real-time video rates. It coordinates data flow between all elements of the video processor as well as data flow to the output frame buffer through the reference frame unit. It is capable of handling both interlaced and noninterlaced video and can operate on both video data streams simultaneously.

Although primarily a sophisticated video and graphics system, the CL-PX2070 may be thought of as one of a growing family of single-chip vision architectures that will be introduced during the mid-1990s. Thus those of us concerned with vision systems and their use should now be aware that a number of single-chip architectures are available that can be combined into image processing systems whose sophistication will rapidly increase while space and cost requirements will become far less than today. This should lead to new vision architectures and new vision systems having unprecedented capabilities at surprisingly low cost.

## Summary and Conclusions

In conclusion we note that, although the basic architectural types (Figure 1) have not changed, their implementation is considerably different today than 30 years ago. In 1961 the single PE of the Perkin-Elmer CELLSCAN required an entire rack. In the 1970s Coulter integrated eight CELLSCAN-type PEs on a single board. In the 1980s a single such PE was reduced to one chip by Johns Hopkins University. Today Martin Marietta has placed 96 PEs on a single chip and is planning hundreds of PEs per chip later in the 1990s.

The major effect of these changes in implementation is to make the cellular automaton the architecture of choice for the most advanced vision systems - especially vision systems requiring real-time execution of tasks consisting of hundreds of steps in both 2D and 3D environments. Thus the cellular automaton is becoming the "vision supercomputer" and replacing the traditional single-PE supercomputer where, as stated at MIT (Hillis, 1985), "A supercomputer inside a six-inch cube would take one nanosecond to send a single signal from one corner of the cube to the other. A nanosecond cycle time is less than a factor of a hundred better than currently available machines." This explains why future generations of vision supercomputers will consist of tens of thousands of PEs.

The cellular automaton architecture has the enormous advantage of having thousands of programmable ALUs

with one dedicated to each pixel or, in the Wavetracer Zephyr to each voxel (volume element).

Although this presents new challenges both to the programmer and to the designer of compilers, it leads to new algorithmic possibilities never before available. Thus, as the 1990s evolve and vision systems using cellular architectures become even less costly with the advent of 100 million transistor chips by the year 2000, this author predicts that automated vision systems that approach the performance of the human eye and brain will come closer to reality.

## References

- Banks ER: Universality in cellular automata. Proc 11th Switch, Automata Th Conf (1970) 216-224.
- Batcher KE: Design of a massively parallel processor. IEEE Trans Comput C-29:836-840 (1980).
- Dinneen GP: Programming pattern recognition. Proc Western Joint Comput Conf, Los Angeles(1955) 94-100.
- Duff MJB and Watson DM: CLIP3: A cellular logic image processor. In *New Concepts and Technologies in Parallel Information Processing* (Caianiello ER, ed), Noordhoff, Leyden (1975) pp 75-86.
- Farmer D, Toffoli T and Wolfram S, eds.: *Cellular Automata*. North-Holland, Amsterdam (1984).
- Fountain TJ: A survey of bit-serial array processor circuits. In *Computing Structures for Image Processing* (Duff MJB, ed.) Academic Press, London (1983) pp 1-14.
- Gardner M: On cellular automata, self-reproduction, the Garden of Eden and the game of 'life'. Sci Amer 224(2):112-117 (1971).
- Golay MJE: Hexagonal parallel pattern transformation. IEEE Trans Comput C-18:733-740 (1969).
- Goldstine HH: *The Computer from Pascal to von Neumann*. Princeton Univ. Press, Princeton, New Jersey (1972).
- Graham MD and Norgren PE: The diff3 analyzer: A parallel/serial Golay image processor. In *Real-Time Medical Image Processing* (Onoe M, Preston KJr, and Rosenfeld A, eds.), Plenum, New York (1980) pp 168-182).
- Gray SB: Local properties of binary images in two dimensions. IEEE Trans Comput C-20(5):551-561 (1971).
- Gray SB: The binary image processor and its applications. Rpt 90365-5C (unpublished), Informational International Inc., Los Angeles (1972).
- Herron JM, Farley J, Preston KJr, and Sellner H: A general-purpose high-speed logical transform image processor. IEEE Trans Comput C-31(8):795-800 (1982).
- Hillis D: *The Connection Machine*, MIT Press (1985).
- Hunt DJ: The ICL DAP and its application to image processing. In *Languages and Architectures for Image Processing* (Duff MJB and Levialdi S, eds.), Academic Press, London (1981) pp 275-282.
- IEEE 75: Proc Intl Symp Uniformly Structured Automata and Logic. CH1052-6C Tokyo (1975).
- Kemeny JG: Man viewed as a machine. Sci Amer 192:58-67 (1955).
- Kruse B: A parallel picture processing machine. IEEE Trans Comput C-22(12):1075-1087 (1973).
- Kruse B: System architecture for image analysis. In *Structured Computer Vision* (Tanimoto S and Klinger A, eds.), Academic Press, New York (1980) pp 169-212.

- McCormick BH: The Illinois pattern recognition computer - ILLIAC III. IEEE Trans Electron Comput EC-12(6):d791-813 (1963).
- McCulloch WS and Pitts W: A logical calculus of the ideas immanent in nervous activity. Bull Math Biophys 5:115-133 (1943).
- Nawrath R and Serra J: Quantitative image analysis: Theory and instrumentation. Microsc Acta 82(2):101-111 (1979).
- Post EL: Finite combinatory processes - Formulation I. J Symb Logic 1:103-105 (1936).
- Preston KJr: The CELLSCAN system - A leucocyte pattern analyzer. Proc Western Joint Comput Conf, Los Angeles (1961) 175-178.
- Preston KJr: Cellular logic arrays for image processing. In *Handbook of Pattern Recognition and Image Processing* (Young TY and Fu K-S, eds.), Academic Press (1986).
- Preston KJr: Abingdon Update. Advanced Imaging (Aug 1992)
- Schrandt RG and Ulam SM: On patterns of growth of figures in two dimensions. N Amer Math Soc 1:642-651 (1960).
- Slotnick DL, Borck, WC, and McReynolds RC: The Solomon computer. Proc Western Joint Comput Conf, Los Angeles, (1962) pp 87-107.
- Slotnick DL: The fastest computer. Sci Amer 224(2):76-87 (1971).
- Smith ARIII: Simple computation - Universal cellular spaces. J Assoc Comput Mach 18:339-353 (1971).
- Sternberg SR: Parallel architectures for image processing. In *Real/Time Parallel Computers* (Onoe M, Preston KJr, and Rosenfeld A, eds.), Plenum, New York (1981) pp 347-359.
- Tojo A: Pattern description with a highly parallel information processing unit. Bull Electrotech Lab 31(8):930-946 (1967).
- Tojo A: Distance functions and minimum path connections. Bull Electrotech Lab 32(9):1930-1942 (1968).
- Tojo A, Yamaguchi T, and Aoyama H: Pattern description with highly parallel information processing unit. VI - Construction and simulation. Bull Electrotech Lab 33(5):479-505 (1970).
- Turing AM: On computable numbers, with an application to the Entscheidungs-problem. Proc London Math Soc, Series 2, 42:230-265 (1936).
- Ulam SM: On some mathematical problems connected with patterns of growth of figures. Proc Symposia Appl Math, Amer Math Soc 11:214-224 (1962).
- Unger SH: A computer oriented toward spatial problems. Proc IRE 46:1744-1750 (1958).
- Yamada H and Amoroso SM: Structural and behavioral equivalences of tessellation automata. Inform Control 18:1-31 (1971).
- Young TY and FU K-S, eds.: *Handbook of Pattern Recognition and Image Processing*, Academic Press (1986).

