Lecture Notes in Computer Science

Edited by G. Goos and J. Hartmanis

109

Digital Image Processing Systems

Edited by Leonard Bolc and Zenon Kulpa

Springer-Verlag
Berlin Heidelberg New York

Lecture Notes in Computer Science

Edited by G. Goos and J. Hartmanis

109

Digital Image Processing Systems

Edited by Leonard Bolc and Zenon Kulpa

Springer-Verlag
Berlin Heidelberg New York 1981

Editors

Leonard Bolc
Institute of Informatics, Warsaw University
PKiN, pok. 850, 00-901 Warszawa, Poland

Zenon Kulpa
Institute of Biocybernetic[redacted]
00-818 Warszawa, Polan[redacted]

AMS Subject Classific[redacted]
CR Subject Classifications (1981): 3.63

ISBN 3-540-10705-3 Springer-Verlag Berlin Heidelberg New York
ISBN 0-387-10705-3 Springer-Verlag New York Heidelberg Berlin

Printing and binding: Beltz Offsetdruck, Hemsbach/Bergstr.
2145/3140-543210

Lecture Notes in Computer Science

Vol. 23: Programming Methodology. 4th Informatik Symposium, IBM Germany Wildbad, September 25–27, 1974. Edited by C. E. Hackl. VI, 501 pages. 1975.

Vol. 24: Parallel Processing. Proceedings 1974. Edited by T. Feng. VI, 433 pages. 1975.

Vol. 25: Category Theory Applied to Computation and Control. Proceedings 1974. Edited by E. G. Manes. X, 245 pages. 1975.

Vol. 26: GI-4. Jahrestagung, Berlin, 9.–12. Oktober 1974. Herausgegeben im Auftrag der GI von D. Siefkes. IX, 748 Seiten. 1975.

Vol. 27: Optimization Techniques. IFIP Technical Conference. Novosibirsk, July 1–7, 1974. (Series: I.F.I.P. TC7 Optimization Conferences.) Edited by G. I. Marchuk. VIII, 507 pages. 1975.

Vol. 28: Mathematical Foundations of Computer Science. 3rd Symposium at Jadwisin near Warsaw, June 17–22, 1974. Edited by A. Blikle. VII, 484 pages. 1975.

Vol. 29: Interval Mathematics. Procedings 1975. Edited by K. Nickel. VI, 331 pages. 1975.

Vol. 30: Software Engineering. An Advanced Course. Edited by F. L. Bauer. (Formerly published 1973 as Lecture Notes in Economics and Mathematical Systems, Vol. 81) XII, 545 pages. 1975.

Vol. 31: S. H. Fuller, Analysis of Drum and Disk Storage Units. IX, 283 pages. 1975.

Vol. 32: Mathematical Foundations of Computer Science 1975. Proceedings 1975. Edited by J. Bečvář. X, 476 pages. 1975.

Vol. 33: Automata Theory and Formal Languages, Kaiserslautern, May 20–23, 1975. Edited by H. Brakhage on behalf of GI. VIII, 292 Seiten. 1975.

Vol. 34: GI – 5. Jahrestagung, Dortmund 8.–10. Oktober 1975. Herausgegeben im Auftrag der GI von J. Mühlbacher. X, 755 Seiten. 1975.

Vol. 35: W. Everling, Exercises in Computer Systems Analysis. (Formerly published 1972 as Lecture Notes in Economics and Mathematical Systems, Vol. 65) VIII, 184 pages. 1975.

Vol. 36: S. A. Greibach, Theory of Program Structures: Schemes, Semantics, Verification. XV, 364 pages. 1975.

Vol. 37: C. Böhm, λ-Calculus and Computer Science Theory. Proceedings 1975. XII, 370 pages. 1975.

Vol. 38: P. Branquart, J.-P. Cardinael, J. Lewi, J.-P. Delescaille, M. Vanbegin. An Optimized Translation Process and Its Application to ALGOL 68. IX, 334 pages. 1976.

Vol. 39: Data Base Systems. Proceedings 1975. Edited by H. Hasselmeier and W. Spruth. VI, 386 pages. 1976.

Vol. 40: Optimization Techniques. Modeling and Optimization in the Service of Man. Part 1. Proceedings 1975. Edited by J. Cea. XIV, 854 pages. 1976.

Vol. 41: Optimization Techniques. Modeling and Optimization in the Service of Man. Part 2. Proceedings 1975. Edited by J. Cea. XIII, 852 pages. 1976.

Vol. 42: James E. Donahue, Complementary Definitions of Programming Language Semantics. VII, 172 pages. 1976.

Vol. 43: E. Specker und V. Strassen, Komplexität von Entscheidungsproblemen. Ein Seminar. V, 217 Seiten. 1976.

Vol. 44: ECI Conference 1976. Proceedings 1976. Edited by K. Samelson. VIII, 322 pages. 1976.

Vol. 45: Mathematical Foundations of Computer Science 1976. Proceedings 1976. Edited by A. Mazurkiewicz. XI, 601 pages. 1976.

Vol. 46: Language Hierarchies and Interfaces. Edited by F. L. Bauer and K. Samelson. X, 428 pages. 1976.

Vol. 47: Methods of Algorithmic Language Implementation. Edited by A. Ershov and C. H. A. Koster. VIII, 351 pages. 1977.

Vol. 48: Theoretical Computer Science, Darmstadt, March 1977. Edited by H. Tzschach, H. Waldschmidt and H. K.-G. Walter on behalf of GI. VIII, 418 pages. 1977.

Vol. 49: Interactive Systems. Proceedings 1976. Edited by A. Blaser and C. Hackl. VI, 380 pages. 1976.

Vol. 50: A. C. Hartmann, A Concurrent Pascal Compiler for Minicomputers. VI, 119 pages. 1977.

Vol. 51: B. S. Garbow, Matrix Eigensystem Routines – Eispack Guide Extension. VIII, 343 pages. 1977.

Vol. 52: Automata, Languages and Programming. Fourth Colloquium, University of Turku, July 1977. Edited by A. Salomaa and M. Steinby. X, 569 pages. 1977.

Vol. 53: Mathematical Foundations of Computer Science. Proceedings 1977. Edited by J. Gruska. XII, 608 pages. 1977.

Vol. 54: Design and Implementation of Programming Languages. Proceedings 1976. Edited by J. H. Williams and D. A. Fisher. X, 496 pages. 1977.

Vol. 55: A. Gerbier, Mes premières constructions de programmes. XII, 256 pages. 1977.

Vol. 56: Fundamentals of Computation Theory. Proceedings 1977. Edited by M. Karpiński. XII, 542 pages. 1977.

Vol. 57: Portability of Numerical Software. Proceedings 1976. Edited by W. Cowell. VIII, 539 pages. 1977.

Vol. 58: M. J. O'Donnell, Computing in Systems Described by Equations. XIV, 111 pages. 1977.

Vol. 59: E. Hill, Jr., A Comparative Study of Very Large Data Bases. X, 140 pages. 1978.

Vol. 60: Operating Systems, An Advanced Course. Edited by R. Bayer, R. M. Graham, and G. Seegmüller. X, 593 pages. 1978.

Vol. 61: The Vienna Development Method: The Meta-Language. Edited by D. Bjørner and C. B. Jones. XVIII, 382 pages. 1978.

Vol. 62: Automata, Languages and Programming. Proceedings 1978. Edited by G. Ausiello and C. Böhm. VIII, 508 pages. 1978.

Vol. 63: Natural Language Communication with Computers. Edited by Leonard Bolc. VI, 292 pages. 1978.

Vol. 64: Mathematical Foundations of Computer Science. Proceedings 1978. Edited by J. Winkowski. X, 551 pages. 1978.

Vol. 65: Information Systems Methodology, Proceedings, 1978. Edited by G. Bracchi and P. C. Lockemann. XII, 696 pages. 1978.

Vol. 66: N. D. Jones and S. S. Muchnick, TEMPO: A Unified Treatment of Binding Time and Parameter Passing Concepts in Programming Languages. IX, 118 pages. 1978.

Vol. 67: Theoretical Computer Science, 4th GI Conference, Aachen, March 1979. Edited by K. Weihrauch. VII, 324 pages. 1979.

Vol. 68: D. Harel, First-Order Dynamic Logic. X, 133 pages. 1979.

Vol. 69: Program Construction. International Summer School. Edited by F. L. Bauer and M. Broy. VII, 651 pages. 1979.

Vol. 70: Semantics of Concurrent Computation. Proceedings 1979. Edited by G. Kahn. VI, 368 pages. 1979.

Vol. 71: Automata, Languages and Programming. Proceedings 1979. Edited by H. A. Maurer. IX, 684 pages. 1979.

Vol. 72: Symbolic and Algebraic Computation. Proceedings 1979. Edited by E. W. Ng. XV, 557 pages. 1979.

Vol. 73: Graph-Grammars and Their Application to Computer Science and Biology. Proceedings 1978. Edited by V. Claus, H. Ehrig and G. Rozenberg. VII, 477 pages. 1979.

Vol. 74: Mathematical Foundations of Computer Science. Proceedings 1979. Edited by J. Bečvář. IX, 580 pages. 1979.

Vol. 75: Mathematical Studies of Information Processing. Proceedings 1978. Edited by E. K. Blum, M. Paul and S. Takasu. VIII, 629 pages. 1979.

Vol. 76: Codes for Boundary-Value Problems in Ordinary Differential Equations. Proceedings 1978. Edited by B. Childs et al. VIII, 388 pages. 1979.

Vol. 77: G. V. Bochmann, Architecture of Distributed Computer Systems. VIII, 238 pages. 1979.

Vol. 78: M. Gordon, R. Milner and C. Wadsworth, Edinburgh LCF. VIII, 159 pages. 1979.

Vol. 79: Language Design and Programming Methodology. Proceedings, 1979. Edited by J. Tobias. IX, 255 pages. 1980.

Vol. 80: Pictorial Information Systems. Edited by S. K. Chang and K. S. Fu. IX, 445 pages. 1980.

Vol. 81: Data Base Techniques for Pictorial Applications. Proceedings, 1979. Edited by A. Blaser. XI, 599 pages. 1980.

Vol. 82: J. G. Sanderson, A Relational Theory of Computing. VI, 147 pages. 1980.

Vol. 83: International Symposium Programming. Proceedings, 1980. Edited by B. Robinet. VII, 341 pages. 1980.

Vol. 84: Net Theory and Applications. Proceedings, 1979. Edited by W. Brauer. XIII, 537 Seiten. 1980.

Vol. 85: Automata, Languages and Programming. Proceedings, 1980. Edited by J. de Bakker and J. van Leeuwen. VIII, 671 pages. 1980.

Vol. 86: Abstract Software Specifications. Proceedings, 1979. Edited by D. Bjørner. XIII, 567 pages. 1980

Vol. 87: 5th Conference on Automated Deduction. Proceedings, 1980. Edited by W. Bibel and R. Kowalski. VII, 385 pages. 1980.

Vol. 88: Mathematical Foundations of Computer Science 1980. Proceedings, 1980. Edited by P. Dembiński. VIII, 723 pages. 1980.

Vol. 89: Computer Aided Design - Modelling, Systems Engineering, CAD-Systems. Proceedings, 1980. Edited by J. Encarnacao. XIV, 461 pages. 1980.

Vol. 90: D. M. Sandford, Using Sophisticated Models in Resolution Theorem Proving.
XI, 239 pages. 1980

Vol. 91: D. Wood, Grammar and L Forms: An Introduction. IX, 314 pages. 1980.

Vol. 92: R. Milner, A Calculus of Communication Systems. VI, 171 pages. 1980.

Vol. 93: A. Nijholt, Context-Free Grammars: Covers, Normal Forms, and Parsing. VII, 253 pages. 1980.

Vol. 94: Semantics-Directed Compiler Generation. Proceedings, 1980. Edited by N. D. Jones. V, 489 pages. 1980.

Vol. 95: Ch. D. Marlin, Coroutines. XII, 246 pages. 1980.

Vol. 96: J. L. Peterson, Computer Programs for Spelling Correction: VI, 213 pages. 1980.

Vol. 97: S. Osaki and T. Nishio, Reliability Evaluation of Some Fault-Tolerant Computer Architectures. VI, 129 pages. 1980.

Vol. 98: Towards a Formal Description of Ada. Edited by D. Bjørner and O. N. Oest. XIV, 630 pages. 1980.

Vol. 99: I. Guessarian, Algebraic Semantics. XI, 158 pages. 1981.

Vol. 100: Graphtheoretic Concepts in Computer Science. Edited by H. Noltemeier. X, 403 pages. 1981.

Vol. 101: A. Thayse, Boolean Calculus of Differences. VII, 144 pages. 1981.

Vol. 102: J. H. Davenport, On the Integration of Algebraic Functions. 1–197 pages. 1981.

Vol. 103: H. Ledgard, A. Singer, J. Whiteside, Directions in Human Factors of Interactive Systems. VI, 190 pages. 1981.

Vol. 104: Theoretical Computer Science. Ed. by P. Deussen. VII, 261 pages. 1981.

Vol. 105: B. W. Lampson, M. Paul, H. J. Siegert, Distributed Systems – Architecture and Implementation. XIII, 510 pages. 1981.

Vol. 106: The Programming Language Ada. Reference Manual. X, 243 pages. 1981.

Vol. 107: International Colloquium on Formalization of Programming Concepts. Proceedings. Edited by J. Díaz and I. Ramos. VII, 478 pages. 1981.

Vol. 108: Graph Theory and Algorithms. Edited by N. Saito and T. Nishizeki. VI, 216 pages. 1981.

Vol. 109: Digital Image Processing Systems. Edited by L. Bolc and Zenon Kulpa. V, 353 pages. 1981.

P R E F A C E

Pictorial information, in all its varieties, constitutes the most important source of our sensory data as well as (apart from the phonetic language) the most general means of communication between people. Inevitably, use of this sort of information becomes steadily the most important means of man-computer communication.

It has started to develop almost at the beginning of computer era: in a sense, the tens of blinking lamps on the early computer panels were a means of visual communication. Apart from this primitive "visual communication", the use of true pictures to exchange information between computers and people can be divided into two main types:

a) real-image processing and analysis
b) computer graphics.

In image processing, the real images from the outside world (real scenes photographs, microscopic images, satellite images, fingerprints, and many others) are inputted to the computer (e.g. by TV means) and processed by it. The results of processing can be of different types: other pictures (e.g. enhanced, noisefiltered, etc.), quantitative descriptions of the picture contents (e.g. number of objects, areas of cells, positions of some features, etc.), recognition decisions (e.g. name of an alphanumeric character, fingerprint classification code, abnormal cell identification, etc.), interpretations (e.g. meaning of a scene, description of a particle-collision event in nuclear physics, etc.). The new use of image processing to store and retrieve pictures in large pictorial data bases is also emerging presently.

In computer graphics, generally not the real images, but descriptions of some, more or less "abstract" drawings are inputted by a human operator to the computer. The input has the character of expressions in some descriptive (artificial) language and/or manual "drawing" (pointing out required positions) with a light-pen on the display screen. The computer stores these picture descriptions in some internal (usually non-pictorial) form and displays them in pictorial form on the graphic display screen (or draws on the ploter) for the convenience of the human operator. It can also introduce some "corrections" to these pictures (e.g. straightening of crooked lines drawn by the light-pen), manipulate them (e.g. zooming, rotation in space) and calculate required parameters (e.g. transmittance of the electronic circuit from the scheme drawn, strain distribution along the beam, etc.). The computer animation of cartoons also uses these techniques. Generally, in image processing the input images are processed by computer (producing eventually some descriptions or "understanding" of their meaning), whereas in computer graphics the images are generated by a computer on the basis of their input descriptions. Both areas share, nevertheless, certain common features, which arise from manipulation of common type of data (pictures) and manifest themselves in the field of picture description and manipulation (for the display).

This book is dedicated to digital systems of image processing. Several European computer systems are described here in detail: GOP and CELLO from Sweden, BIHES ("Budapest Intelligent Hand-Eye-System") from Hungary, CPO-2/K-202 from Poland and S.A.M. (called previously MODSYS) from Federal Republic of Germany.

For various reasons, some other interesting European systems have not been included here. To compensate this, a fairly representative survey of European systems has been included. It reviews and compares systematically eleven systems, including all these listed above. The survey is a somewhat extended and reworked version of an invited paper presented at the EUSIPCO-80 Conference held in Lausanne in September 1980.

In order to show the readers possible practical usefulness of such systems and to introduce them into the methods and techniques of image processing, the book has been augmented finally by the paper by Milgram and Rosenfeld, the leading specialists in the field. This paper presents on a specific example of infrared images analysis a wide range of methods and techniques of image processing, analysis and recognition.

The editors sincerely acknowledge the collaboration of all the contributors to the book and wish to express their gratitude to the European Association for Signal Processing EURASIP for their kind permission to use the survey paper from EUSIPCO-80-Conference for this book.

The authors would like to express their thanks to Springer-Verlag for publishing this volume.

Warsaw, January 1981

Leonard B o l c
Zenon K u l p a

TABLE OF CONTENTS

UNIVERSAL DIGITAL IMAGE PROCESSING SYSTEMS IN EUROPE - A COMPARATIVE SURVEY

by

Zenon KULPA

Polish Academy of Sciences,
Institute of Biocybernetics and Biomedical Engineering
00-818 WARSAW, Poland

Abstract

In the paper, a selected group of eleven universal (computer based) image processing systems is surveyed and compared. They constitute a seemingly representative sample of the vast variety of such systems built in the last decade in European countries. The survey covers systems built for research purposes, either in image processing as such or for some other specific problem area, as well as more practically-oriented ones, including a commercially available routine picture analyzer. An overall classification of their general aims as well as basic parameters and features of their hardware structure, software support and application area is given.

1. Introduction

The purpose of this paper is to cast an overall glance at the vast European scene of universal image processing systems designs. In many different research institutions all over Europe there were designed or are being constructed various such systems, aimed either as tools facilitating basic research in digital picture processing or as practical devices for some more or less specific application. They are often constructed independently of the other analogical constructions existing or planned elsewhere. Their structure and parameters are frequently selected on ad hoc basis or result from specific limitations of chance elements (or building blocks) "just at hand" at the time of the system construction. As it is therefore understandable, they represent a great variety of structures, technical parameters as well as usage modes. Nevertheless, some general features can be found in this variety. The goal of this paper is to put some order in it, providing thus some guidelines for future designers, to help them in their own system development.

Because of rather great number of groups interested in picture processing and building their own systems, it was of course impossible to make this survey fully exhaustive. The main criterion of selection has been simply the familiarity of the author with the system, either personal or through generally accessible scientific literature and a sort of a questionnaire sent to the designers (see Acknowledgments section), or both. Several systems, less known to the author (lacking enough technical data to fill in the tables below) have had to be therefore excluded. For example, it resulted in omitting several seemingly interesting systems developed in FRG (see [6]) about which I have got too fragmentaric informations and too late to be able to collect them for this survey.

Furthermore, only universal systems have been considered, i.e. that easily programmable for different tasks of sufficiently wide problem area. In effect, all of them include such or another programmable digital processor used to process pictures: a general--purpose (mini-)computer or a special hardwired image processor. Finally, all ERTS (Earth Resources Technology Satellite) image processing systems have been deliberately excluded from this survey, as they are a class by itself, having rather specific single source of images and their own specific analysis techniques, emphasizing classification of single pixels described by multispectral data, rather than contextual processing of two-dimensional shapes in the picture.

In spite of this non-exhaustiveness, the small set of only 11 systems surveyed here seems to be in several respects quite representative for the diversity of European image processing systems. I apologize for all omissions of the systems whose features substantiate them to be included in any such survey pretending to be representative. Any system designer confident that his system should have been included here is encouraged to send the system characteristics to the author - it will eventually help in preparation of the next version of the survey. The first version of this paper was presented at the EUSIPCO-80 conference in Lausanne [34], and was also included in the materials of the associated course on Parallel Picture Processing [35]. The materials of this course contain also descriptions of several other image processing systems (mostly of the d-type, see below), not surveyed here.

2. Image processing systems

All systems surveyed here are listed in Table 1. In the text they will be referred to by names given in the first column. Those having no name will be "called" by the first three letters of the name of the laboratory head (see [6, 7]).

The systems can be classified according to their general goals. The following classes are distinguishable:

Table 1. Some European image analysis systems.

Name	Country	Institution	References
CELLO	Sweden	Department of Clinical Cytology, University Hospital, Uppsala	[1 - 4]
(Nag)	FRG*)	Fachbereich Informatik, Universität Hamburg	[5, 6]
(Leb)	USSR	Institute of Information Transmission Problems, Moscow	[7 - 9]
BIHES**)	Hungary	Computer and Automation Institute, Budapest	[10 - 12]
MODSYS***)	FRG	Fraunhofer-Institut für Informations- und Daten- -verarbeitung, Karlsruhe	[13 - 16]
CPO-2	Poland	Institute of Biocybernetics and Biomedical Engineering, Warsaw	[17 - 20]
VIP	Italy	Istituto di Cibernetica del CNR, Arco Felice (Napoli)	[21, 22]
PICAP	Sweden	Department of Electrical Engineering, Linköping University, Linköping	[23 - 26]
CLIP 4	England	Department of Physics and Astronomy, University College, London	[27, 28]
GOP	Sweden	Department of Electrical Engineering, Linköping University, Linköping	[32, 33]
Leitz T.A.S.	France & FRG	I.R.S.I.D. et École des Mines de Paris (license); Ernst Leitz Wetzlar GMBH, Wetzlar, and R. Bosch Fernseh-Anlagen GMBH, Darmstadt (production)	[29 - 31]

*) Federal Republic of Germany; **) Budapest Intelligent Hand-Eye System;

***) Final version has been recently renamed S.A.M. (Sensorsystem for Automation and Measurement).

a) Systems created as tools to investigate some specific scientific problem with computational means: not intended to be multiplied in several copies; the principal goal is to solve the problem rather than to build a system (CELLO, (Nag), (Leb)).

b) Systems created as general purpose (although simple) image processing devices: intended for a wide range of processing tasks; rather research- than application-oriented; eventually with some perspective of building several copies for different users; the principal goal is to build a universal system for research in image processing itself rather than to solve some specific application problem (VIP, CPO-2).

c) Systems intermediate between the two above types: with some specific application in mind (e.g. "robot-eye") but universal enough and serving as a "research prototype" rather than a unique laboratory assembly or a finished production model; the principal goal is to build a fairly universal system, although good for some specific application (BIHES, MODSYS).

d) Systems experimenting with new computer architectures for two-dimensional data processing inherent for image processing: used to gain experience in effectiveness of the proposed set of hardware operations and memory organization; the principal goal is to build an effective and universal new processor rather than a simple and cheap "working" assembly of existing devices (PICAP, CLIP 4, GOP).

e) Commercially available systems for routine picture analysis: universal enough to be usable in sufficiently wide range of different practical tasks but simple enough to be feasible for production and marketing; the principal goal is to cover a wide range of rather simple routine applications, yet worth of automatization, due e.g. to massive amounts of analyses required (Leitz T.A.S.).

How these general systems goals influence specific construction features will be shown in the next three sections, discussing hardware, software and application aspects of the systems.

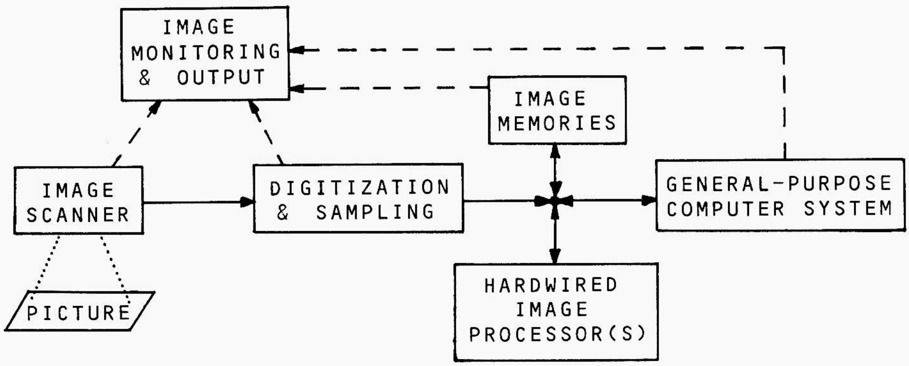

Fig. 1. General scheme of a universal image processing system

3. Hardware structures

The general configuration of a universal image processing system can be schematically drawn as in Fig. 1. Depending on the type of the system as given in the previous section, different parts of this scheme become more important. For the type (a), the central part is the computer in which all processing is done, and the image scanner and output are simply computer peripherals making it possible to input/output the necessary data. They are built (or bought) to fit best the needs of the particular problem. Hardwired processors and image memories are usually absent - the latter are eventually used to extend the computer memory or as flexible display buffers. In the systems of the type (b) and (c), the role of picture input/output and the computer is equalized, image memories are used as input image buffers, hardwired processors are usually absent due to cost considerations and not yet too advanced state of the art. The type (d) systems are built around the proposed hardwired processor. The image memories are partially contained in the processor and partially used as picture input/output buffers. The computer serves as a supervisor facilitating programming and non--pictorial communication with an operator. In the systems of the type (e), all parts are highly integrated, the hardwired processor is an important part, although mainly capable of performing simple counting of picture features on binary images (areas, components, etc.) rather than full-scale of multivalued picture processing.

Table 2. Image input/output devices.

System	Input scanner		Sampling:	Digitization		Output devices
	Type	Scan time	in pixels	Gray levels	Thresholds	
CELLO	OSIRIS: linear diode array mechan. scanned	30-60s	256 × 256	64 (256)	fixed, with software normalization	TV b/w & colour, Versatec
(Nag)	TV	~1s	574 × 512	256	?	COMTAL diplay, Facsimile writer
(Leb)	OPTRONICS P-1700 drum scanner	(large)	max: 1024 × 1024	256	fixed	TV b/w & colour, OPTRONICS P-1700 microfilm plotter
BIHES	TV (vidicon)	20ms	144 × 192	16	off-line control	Tektronix 613 (storage tube)
MODSYS	TV or diode array	20ms	380 × 256	2	fixed	(TV)
CPO-2	TV (vidicon)	40ms	512 × 512	16	hand & computer controlled	TV b/w & colour
VIP	TV (plumbicon)	20-320ms	256 × 256	2 - 16	single threshold computer contr.	Versatec
PICAP	TV	40ms	64 × 64	16	computer contr.	TV b/w & colour
CLIP 4	TV	(20 or 40ms ?)	96 × 96	64	?	TV b/w, Versatec, Tektronix 611, Microfilm plotter
GOP	TV, laser drum scanner	40ms, ?	512 × 512, 4096 × 4096	256	?	TV colour, laser drum plotter
Leitz T.A.S.	TV (plumbicon)	40ms	256 × 256 hexagonal	2: normal or "sliced"	settable to 100 levels	TV colour

The computer (frequently a microprocessor) is used mainly for further elaboration (e.g. statistical) of obtained counts.

In the Table 2 some technical parameters of the image input/output part of the surveyed systems are given. The differences between various types of the systems can be seen quite markedly. The (a)-type systems (CELLO, (Nag), (Leb)) are characterized by usually large number of gray levels (256) and usually special input scanners. For the (Leb) system, its application for image enhancement enforces an accurate and high-resolution image acquisition and hardcopy output. In CELLO, to achieve densitometric accuracy in medical specimens scanning, a special OSIRIS vibrating-prism scanner was adopted (see references in [1, 3]) and the sophisticated light-sensitivity, shading and positioning correction software system run on a dedicated PDP 8/f computer was built [3], see Table 3. In the (Nag) system, however, a standard TV input was applied - it is justified by its use for off-line research in image sequences analysis (e.g. road traffic monitoring), so that less accuracy suffices. The scanning time is also not critical, due to an off-line research-oriented mode of system's usage. It required anyway the storage of sufficiently long image sequences. A large--capacity image memory was needed for this purpose. Here, an analog TV-disk capable of recording 600 TV-frames (Table 3) was employed. All the above systems are therefore characterized by rather slow but usually accurate scanners (except partially (Nag)), great number of pixels scanned (except partially CELLO) and large number of grey levels (usually 256).

All other systems use standard TV camera as an input scanner - it is easily accessible, low-cost, easy in use and accurate enough for general-purpose applications. It is interesting that none of the systems surveyed use flying-spot scanners - they are expensive and limit the form of input pictures, and are used rather in some special-purpose systems (of the (a)-type, e.g. for particle tracks images analysis in nuclear physics). The TV-camera input scanners discussed here (other than (Nag)) sample and digitize the input in real time, i.e. during a single TV-frame, or even half-frame in the case of smaller resolution. The number of pixels is usually small in the (d)-type systems (PICAP: 64 × 64, CLIP 4: 96 × 96). The small number of pixels results from the dimensions of a special parallel or semi-parallel processors array included in these systems -

larger dimensions would result in too big costs and less reliability of the hardware. This small picture "window" can be usually moved over an entire TV-frame and used with different pixel spacing (PICAP). In the CLIP 4, the window is unmovable - the aim of this system is basic research in parallel image processing however, so that an input flexibility has been of secondary importance and made rather simple. The exceptionally high resolution of the GOP system is due to different organization of the processor: the large (up to 512 × 512 pixels) image matrices held in the processor memories are processed by four parallel "computation pipelines", reading the memory in a "rolling" fashion. There is, in fact, no parallel array of processors: the parallelism exists in fetching pixels of the neighbourhood (up to 64 × 64 pixels), but the image scan is performed serially (similar principle was actually applied also in PICAP system).

Practically all systems employ sampling with a square raster. The very interesting exception is the hexagonal raster of the Leitz T.A.S. system. The hexagonal raster has advantages of uniform structure of the local point neighbourhood (no connectivity paradoxes), and smaller number of points in this neighbourhood (25% gain in local operations specification). Its disadvantage, among others, is nonuniform representation of the natural horizontal/vertical coordinate lines [23].

The number of gray levels is usually 16 for most of the non-(a)-type systems, except MODSYS, GOP, Leitz T.A.S., and partially VIP. The latter operate usually on binary (2-level) images. It is much simpler and faster than processing of many-valued images, and it is quite sufficient in a wide range of applications, especially for good quality, high-contrast pictures. Additionally, in many systems the digitization levels can be shifted by the computer, making it possible to discriminate potentially between great many shades of gray (CPO-2, VIP, PICAP, Leitz T.A.S.). For example, in the VIP system, which main aims were simplicity and low cost, a single, but computer-controlled threshold allows an acquisition of images with many (usually 16) gray levels in the course of several input cycles. Similar solution was adopted in the Leitz T.A.S. system, where, additionally, the threshold can be used in a "sliced" manner (setting to the value 1 all pixels between some narrow grey range required).