



Title	Local variable logic operation using birefringent optical elements
Author(s)	Tanida, Jun; Nakagawa, Jun; Ichioka, Yoshiki
Citation	Applied Optics. 1989, 28(16), p. 3467-3473
Version Type	VoR
URL	https://hdl.handle.net/11094/3200
rights	
Note	

The University of Osaka Institutional Knowledge Archive : OUKA

<https://ir.library.osaka-u.ac.jp/>

The University of Osaka

Local variable logic operation using birefringent optical elements

Jun Tanida, Jun Nakagawa, and Yoshiki Ichioka

In this paper we present an optical method for local variable logic operation using birefringence and its extended version for local variable neighborhood operation. Some experimental results verify the validity of the method and show its capabilities. As an example, we consider a simplified case of the method and propose the architecture for a space-variant system suitable for implementing the new method.

I. Introduction

Among the wide range of phenomena concerned with optics, birefringence¹ is useful for polarization-spatial position conversion. The reason polarization-spatial position conversion is of interest is that this conversion technique is useful for the dynamic execution of coded pattern processing with spatial light modulators.

Coded pattern processing is a promising paradigm for optical digital computing, in which data are converted into spatial patterns and processed with pattern manipulation.²⁻⁹ To execute coded pattern processing dynamically, some kinds of spatial light modulator are required. Usually, a spatial light modulator displays 2-D data as the form modulated by polarization. Therefore, if birefringence is efficiently used, we can achieve coded pattern processing itself as well as data coding with a spatial light modulator.

In a previous paper, we demonstrated a novel method of image coding for optical array logic with birefringence.¹⁰ Optical array logic is a technique to execute any logical neighborhood operation for two 2-D binary data in parallel.^{8,9} The important processes necessary for implementing optical array logic are image coding and optical correlation. Optical array logic has many features such as parallelism, flexibility, programmability and so on. Although image coding has been a difficult problem to be solved, the birefringent coding method offers an effective solution.

In addition to the image coding capability, the birefringent scheme enables us to realize local variable logic operation. The method for local variable logic operation is an extension of image coding and can implement sixteen possible logic functions pixel by pixel. The method also can be applied for parallel neighborhood operation as well as pixel-to-pixel operation.

In this paper, we present a method for local variable logic operation using birefringence, and show the modified version of optical array logic for local variable neighborhood operation. Although the modified optical array logic provides flexibility for local variable processing, its efficient use is difficult because of its programming complexity. To avoid this complexity, in this paper we consider a simplified case of the method. Finally, we propose the architecture for a space-variant system capable of implementing the proposed method.

II. Local Variable Logic Operation

The proposed system for local variable logic operation consists of five optical elements, i.e., two controllable halfwave plates (CHPs 1 and 2), two birefringent crystals (BCs 1 and 2), and an optical shutter array. Figure 1 shows an illustrating diagram of local variable logic operation for a corresponding pixel pair, a and b , in the two input images. Pixel data a and b switch the operating mode of CHPs 1 and 2, respectively, in which the value zero causes 90° rotation of the polarizing plane of the incident linearly polarized light; on the other hand, value one does not affect the polarizing state.

BCs split their incident waves into two orthogonally polarized light waves as shown in Fig. 1, where BC 1 and BC 2 split the incident lightwave vertically and horizontally. Therefore, the incident lightwave can reach to any of four different positions and it produces

The authors are with Osaka University, Department of Applied Physics, Suita, Osaka 565, Japan.

Received 7 November 1988.

0003-6935/89/163467-07\$02.00/0.

© 1989 Optical Society of America.

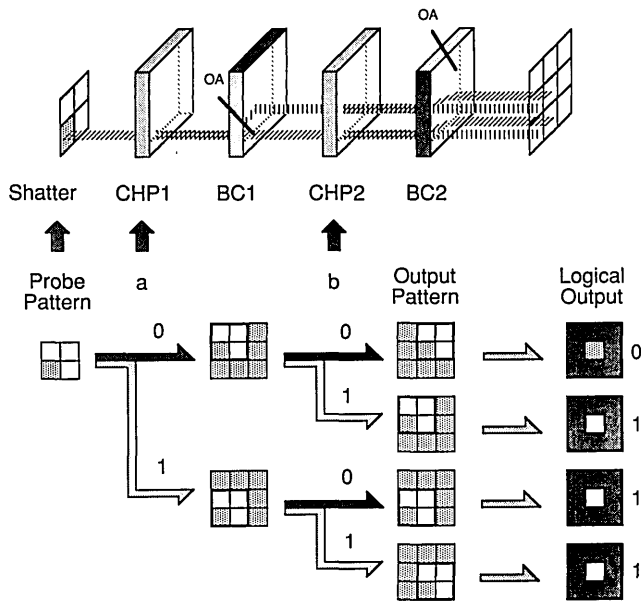


Fig. 1. Schematic diagram of local variable logic operation for a corresponding pixel pair. OA denotes an optical axis of a birefringent crystal.

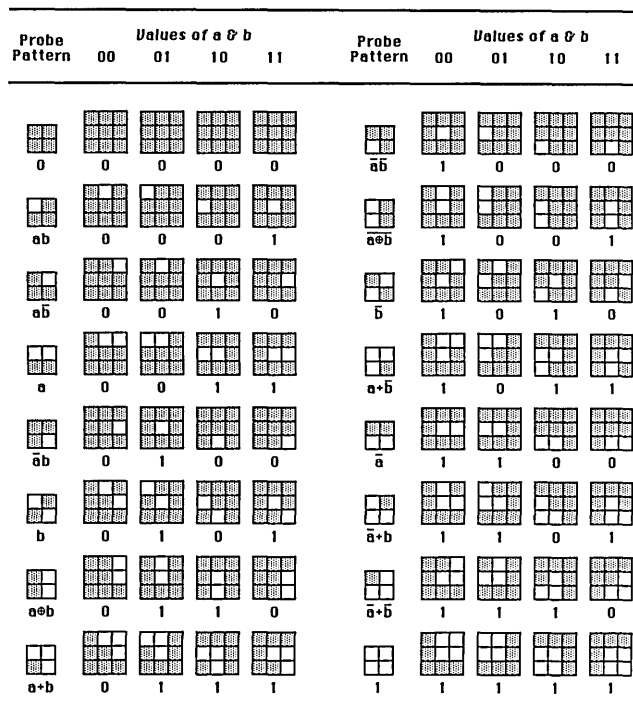


Fig. 2. Probe patterns and the resultant output patterns.

the specific coded patterns according to the combination of values of a and b . Each output pattern is a shifted version of the input pattern specified with an offset value. This situation is shown at the bottom of Fig. 1. The offset value is set to be half the amount of the pixel size.

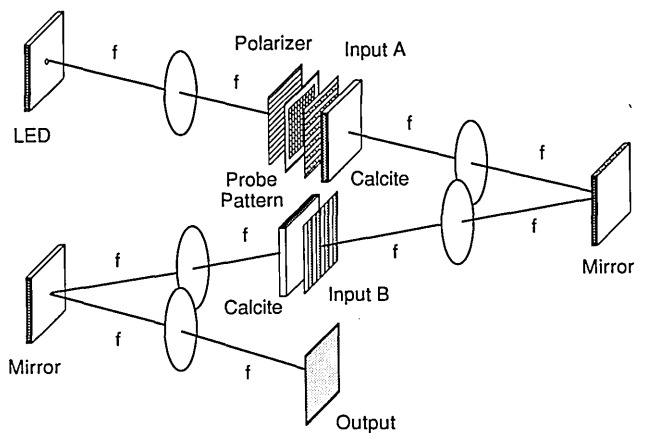
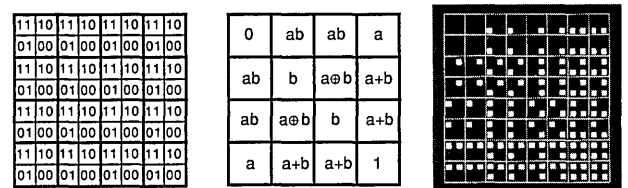


Fig. 3. Experimental setup for a local variable logic operation for two binary images.



(a) Input A&B (b) Logic Functions (c) Probe Patterns

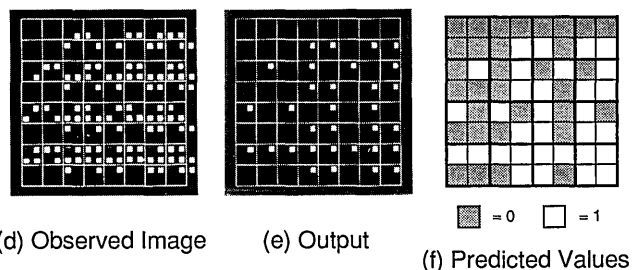


Fig. 4. Experimental result of local variable logic operations: (a) input data of pixel pairs, (b) local functions for pixel pairs, (c) probe patterns, (d) observed locally correlated image, (e) resultant data, and (f) predicted values. To indicate pixel borders, grids are overlaid on (c)–(e).

Let us consider the case using a subpattern consisting of 2×2 pixels as the input. We call this subpattern the probe pattern. As shown in the leftmost column in Fig. 2, sixteen kinds of probe pattern can be considered. Thus, sixty-four possible output patterns can be obtained for the combination of a , b , and the probe pattern. Figure 2 tabulates all of the output patterns. Note that the results are identical to correlated images obtained by shadowcasting scheme³ or optical array logic^{8,9} presented before. From the analogy to both methods, we sample the optical signal at the center part of the output pattern and assign logical value to the sampled signal. Consequently, any binary logical operation for a and b is achieved by selecting the probe pattern. The functions and the logical values noted in

Fig. 2 are shown as those for bright-true logic. Thus, in the case of Fig. 1, OR operation is executed.

While Fig. 1 depicts an operation for a corresponding pixel pair, the same kinds of operations can be implemented in parallel for all pixel pairs over entire input images. In this case, an optical shutter array and arrays of controllable halfwave plates should be prepared. Note that the logical operations for individual pixel pairs can be independently specified by controlling opening of corresponding element shutters in the optical shutter array. Hence, local variable logic operation can be achieved.

Figure 3 shows an experimental setup of local variable logic operation for two binary images. For the lack of appropriate controllable halfwave plates and an optical shutter array, we utilize hollowed halfwave plates as the inputs *A* and *B* and a photographic film for recording probe patterns. They are put in the planes designated by the inputs *A* and *B*. Two mirrors are used to fold the optical system.

Figure 4 shows an experimental result. Inputs are two binary images consisting of 8×8 pixels. They are indicated as the corresponding pixel pairs in Fig. 4(a). The logical operations to be executed for individual pixel pairs are specified as Fig. 4(b). The inputs *A* and *B* are designed so that results of sixteen logical operations for all possible combinations of inputs, 11, 10, 01, 00, can be displayed in different positions in the output plane.

Figure 4(c) is an array of probe patterns overlaid with a grid indicating pixel borders. Figures 4(d) and (e) are the observed locally correlated image and its sampled version, respectively. Grids indicating pixel borders are also overlaid on them. Comparing Fig. 4(e) with the predicted values in Fig. 4(f), it can be verified that correct results by local-variant (space-variant) logic operations have been obtained.

As shown in Fig. 4, no space is needed for separating pixels. Instead, sample points for the output image are set at the upper right of the pixel area (small square area), which correspond to the center part of the output patterns shown in Fig. 1.

III. Extended Version of Optical Array Logic

Although the proposed local variable logic operation method in Sec. II has space-variant processing capability, it cannot communicate signals between any pair of pixels in the input images, namely, an operation for $a_{i,j}$ and $b_{i,j}$ can be executed but that for $a_{i,j}$ and $b_{i,j+1}$ cannot. We found that the method is usable for neighborhood operations making good use of the analogy to optical array logic. We call this method the extended optical array logic.

It can be pointed out that the local variable logic operation is an extension of image coding process in optical array logic. This fact means that the output pattern in Fig. 1 can be used as a coded image in the scheme of optical array logic. Although the interpretation of result obtained is rather complicated, the concept of local and global correlation with local and global kernels helps us to interpret what is the result.

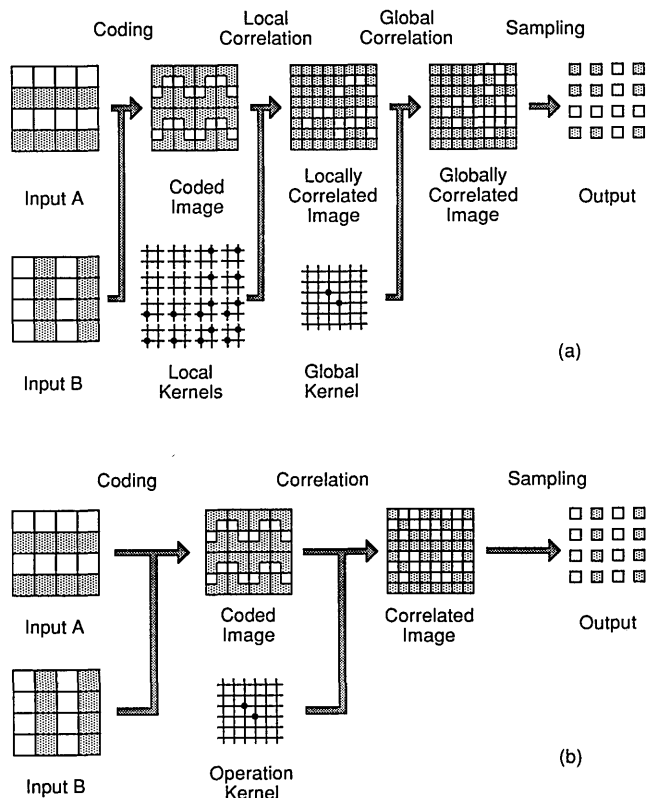
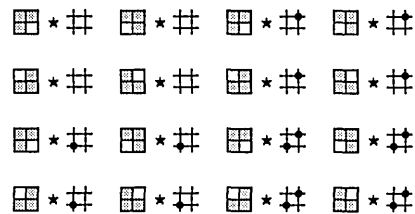


Fig. 5. Schematic diagram of (a) extended optical array logic and (b) original logic.

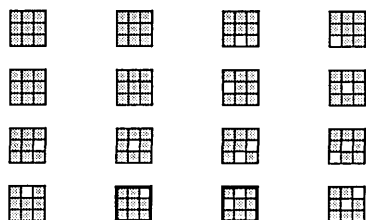
Figure 5(a) shows a schematic diagram of the extended optical array logic useful for explaining the proposed method. For simplicity, we consider a case of one product term operation. In this case, we can thus omit the operation of logical sum after sampling for multiple product term operations. In this scheme, the local correlation of a coded image and a local kernel is added to the original procedure in optical array logic shown in Fig. 5(b); the correlation and the operation kernel in the original scheme are renamed as the global correlation and the global kernel, respectively.

Local correlation is an operation that a coded pattern associated with one pixel pair is correlated with a local kernel (2×2 kernel) specified for the specific pixel pair. Figures 6(a) and (b) explain this situation. Rearrangement of locally correlated patterns for all pixel pairs produces a locally correlated image shown in Fig. 6(c). Adjacent locally correlated patterns overlap each other.

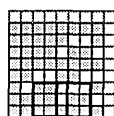
In the extended optical array logic, effective operation kernels synthesized from local and global kernels specify the operation to be executed, whereas, in the original optical array logic, operation kernels alone do that. The procedure to obtain an effective operation kernel from local and global kernels is somewhat complicated. The procedure is described in the Appendix. It should be stressed that any effective operation kernel can be synthesized from given local and global kernels using the procedure. As a result, any logic



(a) Pixel -Divided Correlation



(b) Results of Pixel-Divided Correlation



(c) Locally Correlated Image

Fig. 6. Local correlation processes: (a) pixel-divided correlation, (b) results of pixel-divided correlation, and (c) locally correlated image.

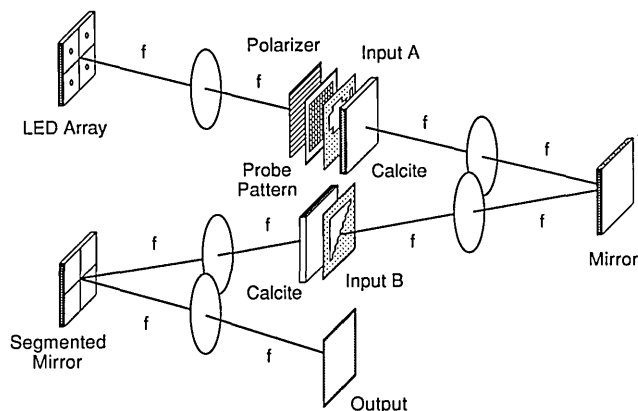
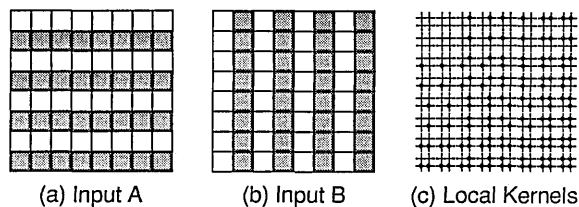


Fig. 7. Possible setup for executing extended optical array logic.

function for neighborhood pixels can be specified and executed with local and global kernels.

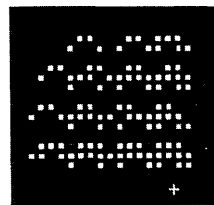
Remember that optical array logic can execute an identical neighborhood operation specified by operation kernels for all pixels in the image. On the other hand, the extended optical array logic can be executed using space-variant operation kernels synthesized from local and global kernels because of space-vari-



(a) Input A

(b) Input B

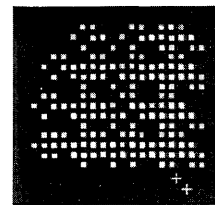
(c) Local Kernels



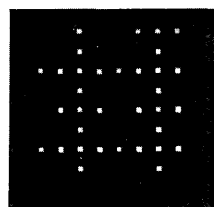
(d) Local Corr. Image



(e) Global Kernel



(f) Global Corr. Image



(g) Sampled Image



(h) Eff.Op.Knl (2,2)



(i) Eff.Op.Knl (4,4)

Fig. 8. Experimental result of local variable logic neighborhood operations: (a) and (b) two input images, (c) array of local kernels, (d) locally correlated image, (e) global kernel, (f) globally correlated image, (g) sampled image, (h) effective operation kernel for (2,2) pixel, and (i) that for (4,4) pixels. Address (0,0) is set at the top left of the image plane.

ance of local kernels. Therefore, local variable neighborhood operation can be specified pixel by pixel with local and global kernels.

Figure 7 shows a possible setup for executing the extended optical array logic. It consists of a local variable logic processor in Fig. 3 and a multichannel reflective correlator.¹⁰ In this system, the local kernels are characterized by the probe patterns on the input A plane, and a global kernel is specified by the tilt angles of mirror pieces in the segmented mirror. The probe pattern is the upside-down version of a local kernel because of the relation between correlation and convolution. While we assume correlation of a coded image and a local kernel in the extended optical array logic, the optical system in Fig. 3 executes convolution of a coded image and a probe pattern.

Note that the optical system using birefringence produces directly a locally correlated image rather than a coded image. After the second calcite in the center arm in Fig. 7, we can obtain the locally correlated image which is correlated with the segmented mirror. The globally correlated image is read out with dark-true logic as well as in the original optical array logic.⁸

Figure 8 is an experimental result of a local variable neighborhood operation. Figures 8(a) and (b) are two input images and Fig. 8(c) indicates an array of local kernels. Figure 8(d) exhibits the locally correlated

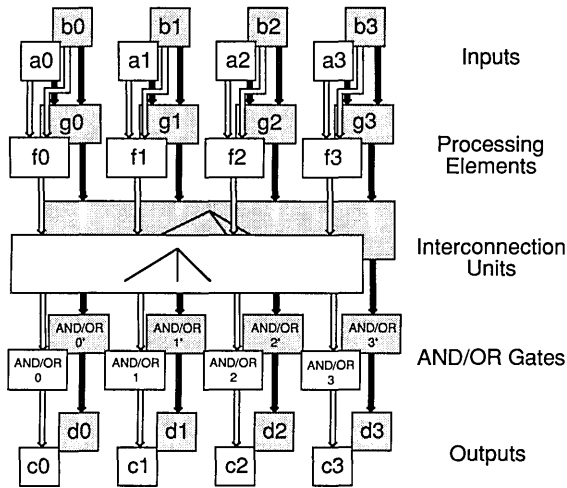


Fig. 9. Possible architecture based on a simplified case of extended optical array logic.

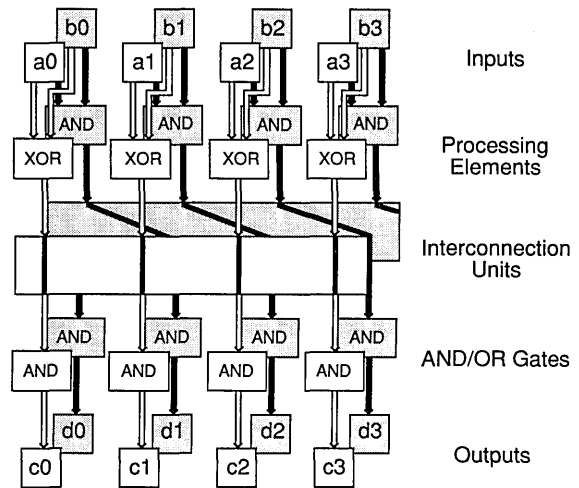


Fig. 10. Example configuration for four-bit adder.

image. Figures 8(e)–(g) show a global kernel, the globally correlated image, and the sampled image, respectively. Figures 8(h) and (i) depict the effective operation kernels for (2,2) and (4,4) pixels, where the origin of the image, (0,0), is set at the top left of the image. The functions for them are

$$c_{2,2} = (\bar{a}_{2,1} + \bar{b}_{2,1})\bar{b}_{2,2}(a_{3,2} + \bar{b}_{3,2}), \quad (1)$$

$$c_{4,4} = (\bar{a}_{4,4} + b_{4,4})(a_{5,4} + b_{5,4}), \quad (2)$$

which are interpreted with kernel unit decomposition described in the Appendix. It can be found that local variable neighborhood operations have been correctly executed.

IV. Possible Architecture for Space-Variant System

In general, designing any kernel capable of implementing desirable operations with a combination of local and global kernels is difficult. This fact increases programming complexity and difficulty in constructing a complex system. In a simplified case, however, we can design local and global kernels for various operations. In the proposed method, a local kernel is used for specifying an operation of a processing element, while a global kernel is employed for determining interconnection patterns between the processing elements. Such function is implemented by global kernels consisting of dots in case 1 in Appendix.

Figure 9 shows a possible architecture, taking account of the above mentioned. In Fig. 9, a_i 's and b_i 's are input data in the inputs A and B, respectively. Processing elements, f_i 's, execute one of sixteen functions specified by corresponding local kernels. The outputs of the processing elements are transferred to output ports through the interconnection unit which interconnects signals with a pattern determined by a global kernel.

The role of AND/OR gates in front of c_i 's is to discriminate dark and bright signals by thresholding. We can select the function of AND or OR for individual process-

ing elements, assigning dark or bright signal to logic value 1; dark-true logic is used for AND operation, while bright-true logic is for OR operation. The outputs of AND/OR gates become the value of c_i 's.

Cascadability of processing is an important requirement for constructing a desirable processing system. For that purpose, we consider the system consisting of dual sets of processing elements, interconnection units, and AND/OR gates. Using this system, we can obtain a pair of outputs, c_i 's and d_i 's, at a time. Each processing set can be independently defined, so that various kinds of data processing are executed.

Figure 10 is one system configuration for a four-bit adder. Although the processor executes only one step of addition, a stack of four processors forms a complete four-bit adder. Of course, optical feedback architecture¹¹ can also be used to reduce total amount of the hardware components.

Technically, the system with this architecture can be constructed with the optical setup shown in Fig. 7. However, this system produces only one output. Therefore, we must consider a method capable of obtaining two outputs as shown in Fig. 9. For this purpose, from the point of the view of compact hardware the system using two wavelengths is expected. Although the system in Fig. 7 does not have feedback loop, use of some spatial light modulators enable us to construct a system with optical feedbacking.

V. Conclusions

We have presented a method for local variable logic operation using birefringence and an extended version of the method for local variable neighborhood operation. Some experimental results have verified the validity of the method and have shown its capabilities. We consider a simplified case of an extended version of optical array logic and propose an architecture suitable for it.

Although only a simplified case was considered in this paper, complex cases must be considered if we want to execute more powerful operations. For that

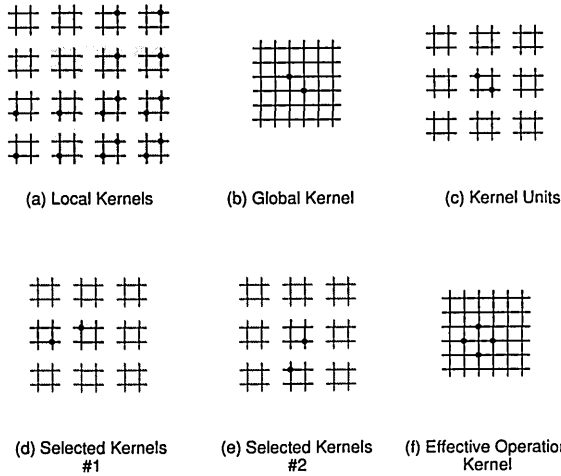


Fig. 11. Synthetic procedure of effective operation kernels: (a) local kernels, (b) global kernel, (c) kernel units decomposed from (b), (d) and (e) selected kernels, and (f) effective operation kernel obtained for (2,2) pixel.

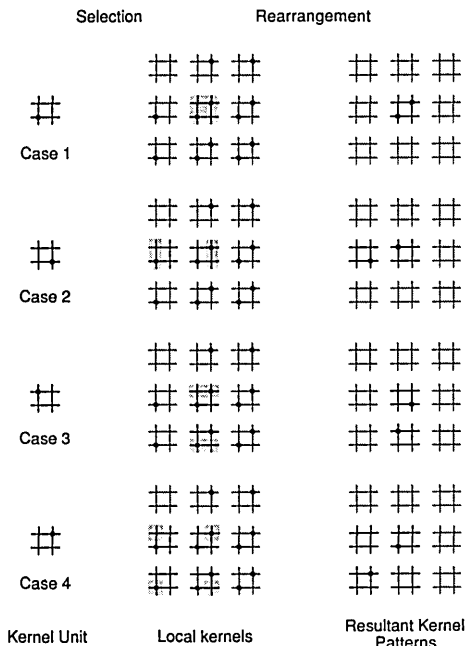


Fig. 12. Selection and rearrangement of kernel patterns. Four cases of dot positions in kernel units are indicated.

purpose, the combined effect of local and global kernels should be analyzed carefully.

Appendix

The synthetic procedure of making effective operation kernels from local and global kernels can be described graphically. We explain it using an example in Fig. 11.

Figures 11(a) and (b) are examples of an array of local kernels and a global kernel. Note that the size of the array of local kernels is the same as that of the image to be processed. Let us consider the operation for (2,2) pixels, where the origin (0,0) is set at the top left of the image. To obtain the effective operation kernel for (2,2) pixels, the following steps are needed:

(1) Decompose the global kernel into kernel units and make a correspondence between the array of local kernels and that of kernel units. In this case, the center of the array of kernel units is set on (2,2) and the shaded area in Fig. 11(a) corresponds to the array in Fig. 11(c). (2) Select kernel patterns from the corresponding local kernels according to the dot position in individual kernel units. Figure 12 shows four cases of the dot position. For case 1, select all dots in the corresponding local kernel. For case 2, select dots in the right side column of the corresponding local kernel and those in the left side column of the next left local kernel. For case 3, select dots in the top row of the corresponding local kernel and those in the bottom row of the next lower local kernel. For case 4, select dots at the four corners of the corresponding next left, next lower, and next lower left local kernels. The dots to be selected are indicated by shading in Fig. 12.

(3) Rearrange the selected kernel patterns according to dot position. For case 1, no rearrange is required. For case 2, exchange left and right columns of individual selected kernels. For case 3, exchange top and bottom rows of individual selected kernels. For case 4, exchange left and right columns and top and bottom rows of individual selected patterns. Right column of Fig. 12 shows the resultant kernel patterns. Figures 11(d) and (e) are the kernel patterns obtained for the example, which are the same as cases 2 and 3 in Fig. 12.

(4) Overlap all the rearranged kernel patterns. Figure 11(f) is the effective operation kernel.

The effective operation kernels obtained can be interpreted by decomposing it into kernel units and us-

Kernel Unit	Function	Kernel Unit	Function
	1		$a + b$
	$\bar{a} + \bar{b}$		$a \oplus b$
	$\bar{a} + b$		b
	\bar{a}		$\bar{a} b$
	$a + \bar{b}$		a
	\bar{b}		$a \bar{b}$
	$\frac{a \oplus b}{2}$		$a b$
	$\bar{a} \bar{b}$		0

Fig. 13. Relation between a kernel unit and function for a pixel pair.

ing the function table shown in Fig. 13. The function executed at (2,2) is

$$c_{2,2} = (\bar{a}_{2,1} + \bar{b}_{2,1})\bar{b}_{2,2}(a_{3,2} + \bar{b}_{3,2}). \quad (A1)$$

Using the same procedure, we can obtain the effective operation kernels for arbitrary pixel pairs.

Note that if a global kernel is composed of kernel units having only a dot as in case 1, its interpretation is easy. In that case, a global kernel is employed for interconnection between pixels. The simplified architecture discussed in Sec. IV is based on this idea.

References

1. M. Born and E. Wolf, *Principles of Optics* (Pergamon, New York, 1975).
2. A. Huang, "Parallel Algorithms for Optical Digital Computers," in *Technical Digest, IEEE Tenth International Optical Computing Conference* (1983), pp. 13-17.
3. J. Tanida and Y. Ichioka, "Optical Logic Array Processor Using Shadowgrams," *J. Opt. Soc. Am* **73**, 800-809 (1983).

4. T. Yatagai, "Optical Space-Variant Logic-Gate Array Based on Spatial Encoding Technique," *Opt. Lett.* **11**, 260-262 (1986).
5. K.-H. Brenner, A. Huang, and N. Streibl, "Digital Optical Computing with Symbolic Substitution," *Appl. Opt.* **25**, 3054-3060 (1986).
6. M. J. Murdocca, "Digital Optical Computing with One-Rule Cellular Automata," *Appl. Opt.* **26**, 682-688 (1987).
7. S. D. Goodman and W. T. Rhodes, "Symbolic-Substitution Applications to Image Processing," *Appl. Opt.* **27**, 1708-1714 (1988).
8. J. Tanida and Y. Ichioka, "Programming of Optical Array Logic. 1: Image Data Processing," *Appl. Opt.* **27**, 2926-2930 (1988).
9. J. Tanida, M. Fukui, and Y. Ichioka, "Programming of Optical Array Logic. 2: Numerical Data Processing Based on Pattern Logic," *Appl. Opt.* **27**, 2931-2939 (1988).
10. J. Tanida, J. Nakagawa, and Y. Ichioka, "Birefringent Encoding and Multichannel Reflective Correlator for Optical Array Logic," *Appl. Opt.* **27**, 3819-3823 (1988).
11. A. Huang, "Design for an Optical General Purpose Digital Computer," *Proc. Soc. Photo-Opt. Instrum. Eng.* **232**, 119-127 (1980).

Books continued from page 3416

focuses on various reconstruction algorithms in transmission and emission tomography. Unlike the other chapters, there are no image reproductions for the techniques discussed, e.g., transform, back projection. It would have been informative to present a visual comparison of the images even if only for the phantom head standard. The last chapter, Magnetic Resonance Imaging and Contrast Enhancement, is by Swenberg and Edward Movius. We gain insight into what can be done to improve image contrast enhancement prior to image capture and electronic processing. To this end, a brief description is given of the major types of MRI contrast agent that have proved useful in modulating the spin relaxation time. The result is that specific tissues would take up these agents, and this in turn would cause an enhancement of MR images.

This book is recommended to optical professionals.

RAYMOND A. HADDAD

The Art of Measurement. Edited by BERNHARD KRAMER. VCH Publishers, New York, 1988. 335 pp. \$66.00.

This volume is a collection of seventeen papers, each a brief survey of recent progress in making more accurate measurements of physical quantities. One of a series of monographs put out by the Physikalisch-Technische Bundesanstalt and edited by J. Bortfeldt, W. Hauser, and H. Rechenberg. The book is divided into three parts: I Fundamental Physics, II Precision Experiments, and III Medicine, and prefaced by a chapter reviewing improvements during the past two centuries in technology and methods for making physical measurements and the resulting increase in our understanding of our physical world.

The part on Fundamental Physics has a paper dealing with the use of lasers for research into areas of basic physics. Laser measurements of frequency shifts verify the deceleration of photons in a gravitational field, the constancy of the velocity of light, and the precise determination of large distances. All results obtained so far indicate that Einstein's theories of special and general relativity are correct. Lasers are also being used to check out some of the basic postulates of quantum mechanics and electrodynamics. Other papers deal with the study of laser properties applying the theory of chaos, fluctuations and localization of properties within materials, quantum mechanics of macroscopic objects (e.g., superconductors) and the use of neutron beam scattering to study properties of con-

densed matter—not accessible or extremely difficult to study by other means.

Part II on Precision Experiments has a paper describing the use of lasers in the very accurate determination of optical properties of atoms, i.e., frequencies and rates for radiative transitions and lifetimes of excited states. Another paper discusses how trapping and cooling single excited ion-atoms may be used to produce very narrow and very stable emission of light for wavelength and frequency standards. Two papers briefly deal with applications of x-rays to interferometry and microscopy. A paper on "Radiometry from the Infrared to the X-Ray Region" describes the use of an electron storage ring as an extremely strong source of x-ray and uv radiation. The acceleration required to maintain electrons in orbit generates radiation in the uv and x-ray regions several orders of magnitude greater than can be obtained from a high temperature blackbody. The final paper in Part II discusses using Josephson junctions and superconducting rings as precision voltage standards and as extremely sensitive detectors of magnetic flux, respectively.

The third and final part of this book on Medicine has just two papers "Lasers in Medicine" and "SQUID-Based Measuring Techniques." In addition to their usefulness in eye surgery, lasers are remarkable tools for microsurgery. Laser beams can be focused to cauterize blood vessels and stop bleeding, and to remove deposits from arteries (angioplasty). In conjunction with appropriate chemical dyes lasers can also be used to destroy cancerous cells without damaging healthy cells surrounding them, and without the after effects associated with techniques of chemotherapy.

The paper on SQUID-based measurements describes how a probe made from a superconducting current loop (the SQUID) can be used to monitor magnetic activity due to individual cells in the brain and other parts of the human body. The SQUID is a tool which may lead to ways of physically locating mental activities within the brain by means of magnetic fields generated by participating neurons. It can also locate currents caused by cardiac activity including those caused by cardiac malfunctioning.

The papers cover a broad range of topics and they can be read and understood by a person with scientific training, without recourse to other reference materials. Since the book is a German publication written by German scientists it may be appropriate to comment that the English is clear with only a few minor difficult or obscure passages.

CHARLES BRAUN

continued on page 3486