Optical Computing Techniques for Image/Video Compression

AKITOSHI YOSHIDA AND JOHN H. REIF, FELLOW, IEEE

Invited Paper

The advantage of optics is its capability of providing highly parallel operations in a three-dimensional space. In this paper, we propose optical architectures to execute various image compression techniques.

We optically implement the following compression techniques:

- *transform coding*
- *vector quantization*
- *video coding*

We show many generally used transform coding methods, for example, the cosine transform, can be implemented by a simple optical system. The transform coding can be carried out in constant time.

Most of this paper is concerned with an innovative optical system for vector quantization using holographic associative matching. Limitations of conventional vector quantization schemes are caused by a large number of sequential searches through a large vector space. Holographic associative matching provided by multiple exposure holograms can offer advantageous techniques for vector-quantization-based compression schemes. Photorefractive crystals, which provide high-density recording in real time, are used as our holographic media. The reconstruction alphabet can be dynamically constructed through training or stored in the photorefractive crystal in advance. Encoding a new vector can be carried out by holographic associative matching in constant time.

An extension to interframe coding using optical block matching methods is also discussed.

Most of the results in this paper were previously presented in our earlier paper [1].

I. Introduction

A. Image Compression

Image compression is crucial for many applications [2]. The objective of image compression is to reduce the bit rate for signal transmission or storage while maintaining an acceptable image quality for various purposes. In video signal transmission, one can compress original images so that the high-quality images which would otherwise require high bandwidth can be transmitted through a medium with relatively low bandwidth. In medical imaging, one can use image compression techniques to store a large number of X-ray pictures that are routinely produced at hospitals.

Compression techniques generally exploit the redundancy in the image. The transform coding such as the Fourier transform coding or the cosine transform coding decomposes an input image into its spectral components [3], [4]. One can achieve compression by appropriately coding the spectral components which are above a certain threshold value. Unfortunately, the difficulties of implementing the transform coding arise from the complexity of transforming algorithms, which require $O(n \log n)$ time or $O(\log n)$ time in parallel for an n-point transformation.

A vector quantizer is a system that maps a set of continuous or discrete vectors into a finite set of discrete vectors that are suitable for transmission or storage. The vector quantization techniques for compressing image and speech signals have been extensively investigated by many researchers [5], [6]. The basic algorithm using full search in the vector space requires an extremely large number of computations. A similar approach using tree search has less computational complexity but requires larger storage and needs still a quite large number of computations.

In video compression, the block matching techniques are used to estimate the displacement of objects [7], [8]. To find the displacement of an object, a small block around the object is correlated with the previous or next frames. The correlation operations require a large number of computations.

These problems are caused by sequential execution of the algorithm on conventional electronic computers. Although some parallelism can be obtained by standard parallel-processing hardware, electrically implemented interconnection may not provide enough bandwidth to handle a large number of computations.

Manuscript received November 1, 1993; revised January 15, 1994. This work was supported by DARPA/ISTO under Grant N00014-91-J-1985, Subcontract KI-92-C-0182, by the NSF under Grant NSF-IRI-91-00681, by NASA under Subcontract 550-63 of prime Contract NAS5-30428, and by US-Israel Binational NSF Grant 88-00282/2. The earlier version of this paper was presented as [1].

The authors are with the Department of Computer Science, Duke University, Durham, NC 27708-0129 USA.

IEEE Log Number 9400879.

0018-9219/94$04.00 © 1994 IEEE

B. Power of Optical Computing

Optical computing has recently become a very active research field [9]–[13]. The obvious advantage of optics is its freedom in space. A set of light beams can establish communication links among optical logic gates in a three-dimensional space, whereas the VLSI model must confine electrical wires on a two-dimensional plane.

From a theoretical computational point of view, for a given problem, there is a lower bound on the circuit area and its computational time. One such lower bound on the planar VLSI model called "AT^2 bounds" states that $AT^2 = \Omega(I^2)$, where A is the circuit area, T is the time used by the circuit, and I is information content[1] of the problem [14]. In a three-dimensional electrooptical model called VLSIO [15], the similar lower bound can be expressed as $VT^{3/2} = \Omega(I^{3/2})$. This implies that as the information content becomes larger, the VLSI circuit requires a larger and larger area to solve the problem in a fixed amount of time. Using three-dimensional optical systems as in the VLSIO model, we can overcome this interconnection problem by utilizing space in a volume.

C. Optics for Image Compression

Although the advantages of optics are well known, the realization of a general-purpose optical computer is yet to happen. On the other hand, special-purpose optical computers have been implemented for areas such as image and signal processing [16], [17], associative memory [18], and neural networks [19], [20]. These systems exploit the advantages of optics such as the ability to perform a two-dimensional Fourier transform in constant time and to implement associative memory using holograms.

A lens can compute the Fourier transform of an input image [21]. A transparency whose transmittance represents the input image is placed at the front focal plane of the lens. The amplitude of the light passing the transparency is modulated by the transmittance. The complex amplitude of the light at the back focal plane represents the Fourier transform of the input image.

Holograms have been used for associative matching [22]–[26]. One can record multiple images on a single holographic medium using distinct reference beams as their associative keys. Later, the stored image can be reconstructed by using its corresponding reference beam as a key associated to the image. Recently, dynamically modifiable holographic media such as photorefractive crystals (i.e., iron-doped $LiNbO_3$) have been widely investigated for associative memory. The refractive index of these media can be optically changed to store holograms. The thickness of the media allows the superposition of many holograms in a common volume in the crystal. A large number of holograms can be stored in a volume by using a recording reference beam that has a distinct angle for each hologram. Later, each hologram can be read out using its corresponding reference beam. Photorefractive crystals are particularly attractive as holographic media, since they provide high-density recording in real time. The maximum number of interconnections that can be implemented in the photocrystal is between 10^8 to 10^{10} /cm^3 [20]. Unfortunately, it is difficult to achieve this limit because of the incoherent erasure during the sequential recording process. The exposure schedule must be carefully controlled to keep the recorded holograms with equal efficiencies. In this case, the maximum number of exposures is in the order of 10^4 [27], [28].

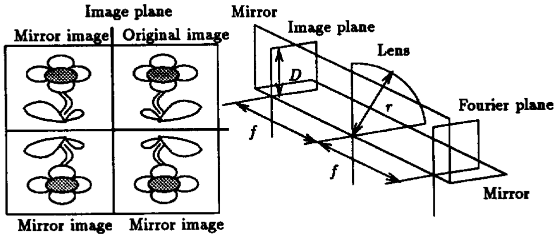


Fig. 1. Optical cosine transform.

II. Optical Cosine Transform Coding

A. Cosine Transform

An optical implementation of a two-dimensional cosine transform is quite simple, as depicted in Fig. 1. A symmetrical image of the original image can be optically created using two mirrors. The two perpendicularly arranged mirrors create three virtual images of the original image. A lens is used to compute a two-dimensional Fourier transform in constant time. The symmetry of the input image allows its cosine transform to be obtained at the back focal plane. The amplitude of each spectral component represents its corresponding Fourier coefficient. Unfortunately, if the intensity is directly measured, we obtain the square of each coefficient. To determine the sign of each coefficient, we can add a constant reference beam at the axis to bias the amplitude of the spectrum components. When the intensity is measured, the amount of this bias term can be subtracted to determine the amplitude of each component.

III. Optical Vector Quantization

A. Vector Quantizer

Extensive studies of vector quantizers (VQ) have been made by many researchers [5], [6]. Lloyd proposed an iterative nonvariational method for optimal PCM design algorithm for scalar variables. Linde, Buzo, and Gray generalized Lloyd's approach to the general vector quantizer design and gave a method referred to as the LBG algorithm [29].

The computational difficulty of VQ comes from the large number of search operations it requires for finding the vector in the codebook that best matches the input vector. To reduce the number of comparisons, Buzo *et al.*

[1] Information content is the number of bits that must cross a boundary in order to solve the problem. The boundary separates the circuit into two sides, each of which holds approximately half the input bits.

introduced a tree-searched vector quantizer (TSVQ). The drawback of TSVQ is that it requires more storage than the full-search VQ and it does not necessarily find the best matching vector.

B. Computing Distortion Measure

It turns out that it is easier to compute the inner product of two vectors in optics. The computation of the squared error distortion is transformed into that of the inner products.

$$d(x,y) = \|x-y\|^2 = \|x\|^2 + \|y\|^2 - 2xy.$$

We introduce two encoding methods.

1) Scaled Coding: Each template vector y is normalized as follows:

$$y'_i = \frac{cy_i}{\|y\|}$$

In this case, the Euclidean norm of the vector is $\|y\|^2 = kc^2$. In Gain/Scale VQ [30], the template vectors are normalized in this way. The input vector is matched against each template vector. The template vector which maximizes the value of the inner product is chosen as the shape vector. The gain factor, which is the value of the inner product, is independently quantized.

2) Expanded Coding: Each input vector x and template vector y, each of length k, are transformed into vectors x' and y', each of length $3k$, respectively, as follows:

$$x'_i = \begin{cases} x_i, & \text{if } i \le k \\ c^2/2 - x_i^2/2, & \text{if } k < i \le 2k \\ 1, & \text{otherwise} \end{cases}$$

$$y'_i = \begin{cases} y_i, & \text{if } i \le k \\ 1, & \text{if } k < i \le 2k \\ c^2/2 - y_i^2/2, & \text{otherwise.} \end{cases}$$

In this case, taking the inner product of x' and y' computes the squared error distortion.

$$\begin{aligned} x'y' &= \sum_i^k x_i y_i + \frac{1}{2}\sum_i^k (s^2 - x_i^2) + \frac{1}{2}\sum_i^k (s^2 - y_i^2) \\ &= kc^2 - \frac{1}{2}(\|x\|^2 + \|y\|^2 - 2xy) \end{aligned}$$

These encoding schemes can be implemented using SLM's or nonlinear optical filters. In the following subsections, we assume one of the above encoding schemes.

C. Design of Holographic VQ

1) General Configuration: Let the size of each block be $\sqrt{k} \times \sqrt{k}$. We use two light source arrays at the input plane: one for the input image block array S_1 and the other for the label array S_2. The first array, S_1, is of size $\sqrt{k} \times \sqrt{k}$ and is used to represent each vector of the input image. Each pixel of the input vector can be represented by either amplitude (coherent reading) or intensity (incoherent reading). For recording, a coherent source must be used, but for reading, either a coherent or incoherent source may be used. If the pixel values are encoded in amplitude, the total intensity of the input vector in Scaled coding method is kept constant. Otherwise, this issue is not very significant to the general configuration. We assume the intensity encoding with incoherent reading.

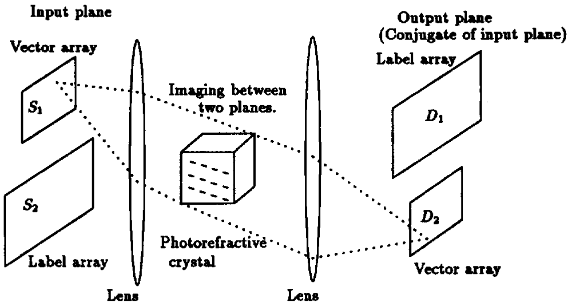


Fig. 2. Holographic vector quantizer.

The other array at the input plane, S_2, is of size $\sqrt{N} \times \sqrt{N}$, and provides one of N label beams to the holographic medium. We have two arrays of photodetectors at the output plane: one for the reconstruction vector array D_1 and the other for the label array D_2. The detector array, D_1, is of size $\sqrt{k} \times \sqrt{k}$ and is used for reconstructing the decoded vector. The other array, D_2, is of size $\sqrt{N} \times \sqrt{N}$ and is used to detect the best matching label for the input vector. Figure 2 shows a configuration of the system. Two Fourier lenses are used to form Fourier transform holograms in the photorefractive crystal placed between the lenses.

We consider each pixel in the input plane as a point source. Each point source produces a plane wave after passing the first lens. There are k pixels in S_1 and N pixels in S_2. At the output plane, there are k pixels in D_1 and N pixels in D_2. We give a global index number to each pixel of the vector arrays and the label arrays. The pixels of the vector arrays are indexed from 1 to k, and those of the label arrays are indexed from $k+$ 1 to $k + N$. The image of the ith pixel in the input plane is formed at the position of the ith pixel in the output plane via the two Fourier lenses.

In photorefractive crystals, holograms are recorded via the modulation of the index of refraction by the space-charge field induced by a spatially varying optical intensity distribution. Interaction between light waves and multiple gratings in the volume can be treated as linear if the diffraction efficiencies of the gratings are very small [31]. In this case, each incident plane wave interacts with every grating in the volume independently. The diffracted light for an arbitrary configuration of the input pixels is thus computed as a sum of each diffracted wave via each grating. Each grating formed by recording a pair of two plane waves is capable of connecting the pair of pixels.

Let η_{ij} be the diffraction efficiency of the grating connecting the ith pixel and the jth pixel. Assuming incoherent reading with the first-order crosstalk, the intensity of light at the jth pixel due to a light wave from the ith pixel with

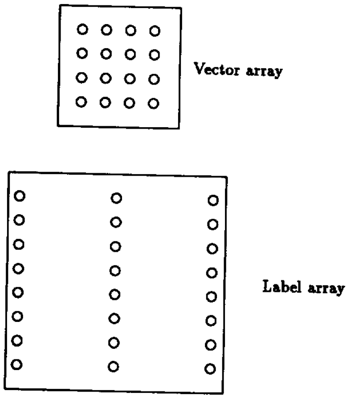


Fig. 3. Pixel arrangement with no crosstalk.

intensity I_i is given by [32]

$$I_j = I_i \eta_{ij} + \sum_{p \neq i} \sum_{q \neq j} I_i \eta_{pq}(i,j)$$

where $\eta_{pq}(i,j)$ denotes the crosstalk diffraction efficiency, which is the diffraction efficiency of light from the ith pixel to the jth pixel due to a grating connecting the pth pixel and the qth pixel.

The phase mismatched crosstalk term can be eliminated under several conditions [31], [32]. The first condition limits the minimum spacing between pixels in the vertical direction. In other words, the minimum spacing of two pixels must be greater than $\lambda f/L$, where f is the focal length of the lens. The second condition restricts the placement of pixels in the horizontal direction. A grating that connects two pixels in a column may connect the pixels in the same two rows of other columns. Thus one way to satisfy this condition is to keep the horizontal spacing between label pixels larger than the whole dimension of the vector array as in Fig. 3.

Under these assumptions, a hologram connecting a pair of pixels can be recorded without seriously affecting the connections between other pairs. If we assume the phase mismatched crosstalk term to be negligible, we can consider the photorefractive crystal as a linear mapping from input plane vector I to output plane vector O. I is a vector of length $k + N$ whose elements correspond to the intensity values of pixels at the input plane. Similarly, O is a vector of length $k+N$ whose elements correspond to the intensity values of pixels at the output plane. The mapping from the input plane to the output plane can be written as a matrix vector multiplication.

$$O = HI$$

where H is a $(k + N) \times (k + N)$ matrix whose (i,j)th entry is η_{ij}.

In this paper, we assume the above model to characterize photorefractive crystals. However, the nonlinear dynamics of multiple-exposure holograms make it extremely difficult to characterize analytically the recorded holograms. It is not possible to record multiple holograms independently. The exposure of each new hologram partially erases previously recorded holograms. For a more complete analysis, one may have to rely on experiments.

2) Encoding and Decoding: We assume that the reconstruction alphabet of size N is already stored in the photorefractive crystal. The photorefractive crystal is initialized by a uniform plane light wave. To record the reconstruction alphabet, each vector in the reconstruction alphabet is loaded into array S_1 one at a time. Array S_2 can produce one of the N label beams.

The exposure of a pair of each vector wave and its associated label beam increases the connection strength by an amount proportional to the intensity of each pixel. The photorefractive crystal stores each η_{ij} as an element in matrix H[20]. Each vector y_i is of length k and is represented by intensity. Each diffraction efficiency η_{ij} is proportional to the product of the intensity values of the ith pixel and the jth pixel. Thus after recording each vector in this alphabet, we have

$$H \propto \begin{pmatrix} H_{11} & y_1 & y_2 & \cdots & y_N \\ y_1^t & & & & \\ y_2^t & & & I & \\ \cdots & & & & \\ y_N^t & & & & \end{pmatrix}$$

where y_i^t denotes the transposed vector of y_i, and H_{11} is the sum of every outer product of y_i.

Consider encoding a new vector y. The input plane vector I can be written as $I = (y^t, 0, \cdots, 0)$. Thus the output plane vector O can be written as

$$O = ((H_{11} y^t)^t, y_1^t y, y_2^t y, \cdots, y_N^t y).$$

The first k elements of O correspond to the intensity values at D_1 and are not of interest to us. The rest of the N elements correspond to the intensity values at label array D_2. Each pixel receives light with intensity proportional to the inner product of the input vector and the recorded vectors.

In Scaled coding, maximizing the inner product gives the best matching shape vector and quantizing the value of the inner product gives the gain factor. In Expanded coding, the inner product gives the complement of the actual squared error distortion. Thus maximizing the inner product gives the best matching vector. In both coding methods, we choose the label of the pixel with the maximum intensity value as the encoded symbol.

Thus encoding a new vector is simple and fast. There are no explicit computations or searches required to find the best matching vector in the reconstruction alphabet. Each new vector can be loaded into array S_1, which in turn illuminates the photorefractive crystal to retrieve its associated label. The retrieved light reaches some pixels at array D_2. Array D_2 selects the pixel with the maximum intensity and transmits its label as the encoded symbol.

Decoding is very simple. For a given label, the reconstruction of its associated image vector is straightforward. The label is loaded into array S_2 that will illuminate the photorefractive crystal. By associative matching of the

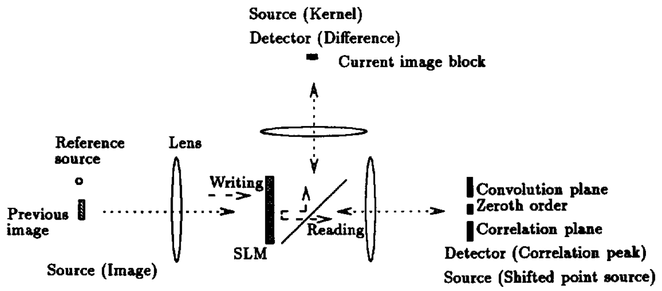


Fig. 4. Block matching by optical correlation.

holograms, the image vector associated with the label will be obtained at array D_1.

The speed of the system is only limited by the switching speed of the source array.

3) Codebook Generation: We consider the LBG algorithm for optimizing the codebook and the splitting technique for creating the initial codebook. In order to implement these algorithms, we need three operations: computing the distortion, partitioning the training sequence into the minimum distortion partition, and computing the centroid of each partition.

Each vector from the training sequence is loaded into array S_1 one at a time. Array S_1 illuminates the photorefractive crystal. The diffracted waves from the crystal are detected by array D_2. The distortion is obtained by measuring the intensity and the training sequence is partitioned into the minimum distortion partition. The centroid of each partition is computed by recording each pattern in the partition.

IV. Optical Video Compression

A. Motion Detection

The block-matching algorithm uses a two-dimensional correlation function to estimate the displacement of small blocks. The computational cost of correlation is high. Thus in practice, each correlation operation is applied for a small block within a relatively small surrounding area.

B. Design of Block Correlator

We consider coding of the translation of each block in the image frame. We use a dynamic cross-correlator to track the movement in the image. There are two types of optical correlators: the matched filtering correlator [33] and the joint transform correlator [34], [35]. The matched filtering correlator requires a preparation of a Fourier hologram of the input image. The input key is used to illuminate the hologram, from which the correlation image as well as the convolution image can be obtained. The joint transform correlator places the input image and the key at the same input plane. The Fourier transform of the light distribution at the input plane is recorded in a hologram. By illuminating the hologram by a plane wave, the autocorrelation image can be obtained. From the autocorrelation image, the correlation image can be extracted. The high-resolution spatial light modulators (SLM's) can be used to implement dynamic correlation operations.

In the matched filtering correlator, the input image is recorded in a holographic medium as a Fourier hologram. When the hologram is illuminated with an input key, the correlation and convolution images can be obtained as the reconstructed waves.

In the Joint Transform correlator, the input image and input key are placed at the same input plane. The hologram records the Fourier power spectrum of the light distribution of the input plane. Since the Fourier transform of the Fourier power spectrum is the autocorrelation function of the original input, a plane wave is used to read the recorded hologram to obtain its Fourier transform at the detector plane. By placing the input image and input key with an appropriate displacement, the correlation of the input key to the input image can be obtained as part of the autocorrelation image.

We describe our interframe coding method using an optical correlator as shown in Fig. 4. For the spatial domain, either VQ or cosine transform methods can be used. First, we consider the matching of a single block. Later, we consider the matching of all the blocks in parallel.

Suppose the previous frame is already encoded. First, the Fourier transform of the previous frame is formed on the SLM and stored. Next, the current frame is loaded into the source array. To encode a block from the current frame, the Fourier transform of the block is superimposed on the SLM. The light reflecting from the SLM forms the cross-correlation of the block with the previous image on the correlation plane. The position of the correlation peak represents the position of the matching block. The next step is to determine the differential block that represents the difference between the matched block and the source block. To obtain the differential block, the correlation peak is used to address the SLM again. When a point source located at the position of the correlation peak is used to address the SLM, the image recorded in the SLM is reconstructed on the current frame array with the displacement equal to that of the point source from the center of the correlation plane. Thus the previous frame is superimposed on the current frame with the correct displacement that matches the matched block and the source block. This allows differences

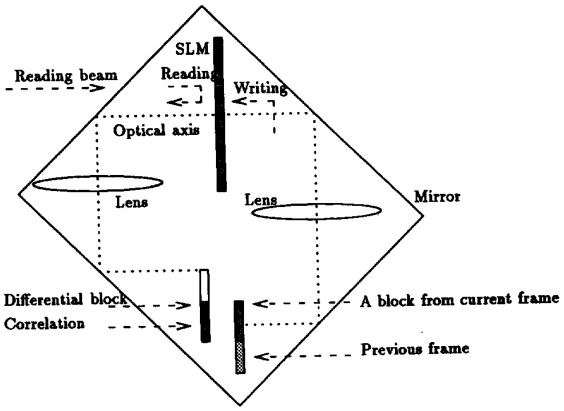


Fig. 5. Block matching by Joint transform optical correlation.

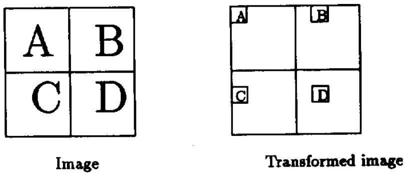


Fig. 6. Transforming blocks for parallel correlation.

in the pixel values of the source block and the matched block to be detected. The displacement vector and the pixel values of the differential block are used to encode the source block.

The block correlator finds the displacement vector and the pixel values of the differential block for each source block sequentially. The current frame is encoded as the set of displacement vectors and differential blocks.

We can use the Joint transform optical correlator for block matching. The block-matching system is depicted in Fig. 5.

In this method, there are two recording phases for each block-matching operation. In the first recording phase, the previous frame and a block from the current frame are placed at the same source array. The light from the source array forms its Fourier transform on the SLM, and it is stored in the SLM as the Fourier power spectrum. A plane wave is used to read the SLM. The Fourier transform of the light distribution at the SLM is formed on the detector array. The detector array finds the correlation peak and the position of the peak is transferred to the source array. The SLM is reset for the second recording phase. In the second recording phase, the previous frame and a point source placed at the position of the correlation peak are used to illuminate the SLM. The SLM stores the power spectrum and the plane wave is used to read the SLM. The light from the SLM forms the previous frame with the correct displacement so that the differential block is obtained at the detector array.

The previous two correlators compute the correlation of a single block. If we wish to obtain the correlation of all the blocks in parallel, we may do so by increasing the hardware complexity. The current image is divided into blocks as before. Then, they are demagnified and transformed into smaller blocks with positional offsets. Figure 6 illustrates this transformation. Blocks are separated by sufficient distance, so that the correlation of each block to the frame is obtained without overlapping.

V. Discussion

We introduced optical techniques for transform coding, vector quantization, and video coding.

The advantage of using a lens to compute a cosine transform is its speed. The computational time does not depend on the size of the input image. The drawback is its analog nature which requires detectors with a large dynamic range.

The advantages of our holographic vector quantizer are its speed and small size. Encoding a vector is carried out in constant time. The scalability of the system is determined by the dynamic range of the photocrystal rather than the diffraction and aberration of the optical system. Although $LiNbO_3$ has large dynamic range to allow a number of multiple exposures, a large number of exposures reduces the dynamic range of each image.

The advantage of using an optical correlator for video coding is its speed. Optical correlation techniques have been used in real-time target analysis. The optical system to perform block-matching operations is not complicated. The limitation is its limited dynamic range. The dimension of the block and the range of pixel values are limited by the dynamic range of the detectors.

References

[1] J. H. Reif and A. Yoshida, "Optical techniques for image compression," in *Proc. IEEE Data Compression Conf. '92*, J. A. Storer, Ed. Norwell, MA: Kluwer, 1992, pp. 32–41.

[2] A. K. Jain, "Image data compression: A review," *Proc. IEEE*, vol. 69, pp. 349–389, 1981.

[3] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," *IEEE Trans. Comput.*, vol. C-23, pp. 90–93, 1974.

[4] R. J. Clarke, *Transform Coding of Images*. London, UK: Academic Press, 1985.

[5] R. M. Gray, "Vector quantization," *IEEE ASSP Mag.*, pp. 4–29, Apr. 1984.

[6] N. M. Nasrabadi and R. A. King, "Image coding using vector quantization: A review," *IEEE Trans. Commun.*, vol. COM-36, pp. 957–971, 1988.

[7] J. R. Jain and A. K. Jain, "Displacement measurement and its application in interframe image coding," *IEEE Trans. Commun.*, vol. COM-30, pp. 1799–1806, 1981.

[8] H. G. Musmann, P. Pirsch, and H.-J. Grallert, "Advances in picture coding," *Proc. IEEE*, vol. 73, no. 4, pp. 523–548, 1985.

[9] J. W. Goodman, F. J. Leonberger, S. Kung, and R. A. Athale, "Optical interconnections for VLSI systems," *Proc. IEEE*, vol. 72, pp. 850–866, 1984.

[10] A. A. Sawchuk and T. C. Strand, "Digital optical computing," *Proc. IEEE*, vol. 72, pp. 758–779, 1984.

[11] T. E. Bell, "Optical computing: A field in flux," *IEEE Spectrum*, vol. 23, no. 8, pp. 34–57, 1986.

[12] D. Feitelson, *Optical Computing, A Survey for Computer Scientists*. Cambridge, MA: MIT Press, 1988.

[13] A. D. McAulay, *Optical Computer Architectures*. New York: Wiley, 1991.

[14] J. D. Ullman, *Computational Aspects of VLSI*. Rockwille, MD: Computer Sci. Press, 1984.

[15] R. Barakat and J. H. Reif, "Lower bounds on the computational efficiency of optical computing systems," *Appl. Opt.*, vol. 26, pp. 1015–1018, 1987.

[16] K. Preston, *Coherent optical computers*. New York: McGraw-Hill, 1972.

[17] F. T. S. Yu, *Optical information processing*. New York: Wiley, 1983.

[18] T. Kohonen, *Self-Organization and Associative Memory*, 2nd ed. New York: Springer-Verlag, 1988.

[19] N. Farhat, D. Psaltis, A. Prata, and E. Paek, "Optical implementation of the Hopfield model," *Appl. Opt.*, vol. 24, pp. 1469–1475, 1985.

[20] D. Psaltis, D. Brady, and K. Wagner, "Adaptive optical networks using photorefractive crystals," *Appl. Opt.*, vol. 27, pp. 1752–1759, 1988.

[21] J. W. Goodman, *Introduction to Fourier Optics*. New York: McGraw-Hill, 1968.

[22] H. J. Caulfield, "Associative mapping by optical holography," *Opt. Commun.*, vol. 55, pp. 80–82, 1985.

[23] H. Kang, C. X. Yang, G. G. Mu, and Z. K. Wu, "Real-time holographic associative memory using doped $LiNbO_3$ in a phase-conjugating resonator," *Opt. Lett.*, vol. 15, pp. 637–639, 1990.

[24] Y. Owechko, G. J. Dunning, E. Marom, and B. H. Soffer, "Holographic associative memory with nonlinearities in the correlation domain," *Appl. Opt.*, vol. 26, pp. 1900–1910, 1987.

[25] B. H. Soffer, G. J. Dunning, Y. Owechko, and E. Marom, "Associative holographic memory with feedback using phase conjugate mirrors," *Opt. Lett.*, vol. 11, pp. 118–120, 1986.

[26] A. Yariv, S. Kwong, and K. Kyuma, "Demonstration of an all-optical associative holographic memory," *Appl. Phys. Lett.*, vol. 48, pp. 1114–1116, 1986.

[27] W. J. Burke, D. L. Saebler, W. Phillips, and G. A. Alphonse, "Volume phase holographic storage in ferroelectric crystals," *Opt. Eng.*, vol. 17, pp. 308–316, 1978.

[28] J. H. Hong, P. Yeh, D. Psaltis, and D. Brady, "Diffraction efficiency of strong volume holograms," *Opt. Lett.*, vol. 15, pp. 344–346, 1990.

[29] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Commun.*, vol. COM-28, pp. 84–95, 1980.

[30] A. Buzo, A. H. Gray, R. M Gray, and J. D. Markel, "Speech coding based upon vector quantization," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-28, pp. 562–574, 1980.

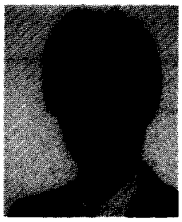
[31] H. Lee, "Volume holographic global and local interconnecting patterns with maximal capacity and minimal first-order crosstalk," *Appl. Opt.*, vol. 28, pp. 5312–5316, 1989.

[32] H. Lee, X. Gu, and D. Psaltis, "Volume holographic interconnections with maximal capacity and minimal cross talk," *J. Appl. Phys.*, vol. 65, pp. 2191–2194, 1989.

[33] A. VanderLugt, "Coherent optical processing," *Proc. IEEE*, vol. 62, pp. 1300–1319, 1974.

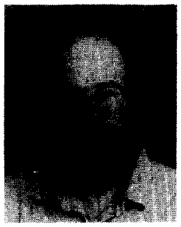
[34] D. Casasent, "Coherent optical pattern recognition," *Proc. IEEE*, vol. 67, pp. 813–825, 1979.

[35] C. S. Weaver and J. W. Goodman, "A technique for optically convolving two functions," *Appl. Opt.*, vol. 5, pp. 1248–1249, 1966.



Akitoshi Yoshida received the B.S. degree in mathematics from Keio University, Japan, in 1988, and the M.S. and Ph.D. degrees in computer science from Duke University, Durham, NC, in 1991 and 1994, respectively.

He is currently at the Department of Computer Science, Duke University. His research interests include image compression, optical computing, and virtual reality.



John H. Reif (Fellow, IEEE) received the B.S. degree in applied mathematics and computer science from Tufts University, Medford, MA, in 1973, and the M.S. and Ph.D. degrees in applied mathematics from Harvard University, Cambridge, MA, in 1975 and 1977, respectively.

He is currently a professor in the Duke University, Durham, NC, Computer Science Department, where he works on the development and analysis of algorithms (particularly parallel algorithms) for various fundamental problems, such as solution of sorting, graph problems, algebraic and numerical problems, and the like. He has implemented sophisticated algorithms on existing massively parallel machines for sorting, data compression, solution of sparse linear systems, and molecular dynamics. He is co-architect and co-inventor of the BLITZEN 128 processor chip, and president of RSIC, Inc., which recently developed special-purpose massively parallel hardware for very high rate lossless data compression. He has served as a consultant to IBM, GTE Laboratories, Thinking Machines, Microsoft, NASA Goddard, MRJ of Park and Elmer, and the Microelectronics Center for North Carolina.

Dr. Reif is a Fellow of the Institute of Combinatorics.