PERIODICA POLYTECHNICA SER. EL. ENG. VOL. 38, NO. 2, PP. 125-145 (1994)

FAST IMPLEMENTATION TECHNIQUES OF MULTICHANNEL DIGITAL FILTERS FOR COLOR IMAGE PROCESSING USING MATRIX DECOMPOSITIONS

B. G. MERTZIOS* and A. N. VENETSANOPOULOS**

* Department of Electrical and Computer Engineering
Democritus University of Thrace
67 100 Xanthi, Hellas
Fax: (30)-541-26473, 26947,
e-mail: mertzios@demokritos.cc.duth.gr

** Department of Electrical and Computer Engineering
University of Toronto
Toronto M5S 1A4, Canada
Fax: (416) 978-4425, 978-7423
e-mail:anv@dsp.toronto.edu

Received: Aug. 10, 1994.

Abstract

For the processing of color images, multivariable 3-input, 3-output 2-D digital filters are used, considering decomposition in the R, G and B components. Assuming that the three image components are decorrelated, three independent single-input, single-output (SISO) two-dimensional (2-D) digital filters are needed for the processing of each monochromatic image. Additional processing is needed for the correlated noise components in each channel. The requirement of very fast processing dictates the use of special purpose hardware implementations. The *VLSI array processors*, which are special purpose, locally interconnected computing networks, are ideally suited for the fast implementation of digital filters, since they maximize concurrency by exploiting both parallelism and pipelining. In this paper fast implementation architectures of 3-input, 3-output 2-D multi-input digital filters for color image processing that are based on matrix decompositions are presented. The resulting structures are modular, regular, have high inherent parallelism and are easily pipelined, so that they may be implemented via VLSI array processors.

Keywords: color image processing, matrix decompositions, fast implementation techniques, parallel architectures, digital filtering.

1. Introduction and State of the Art

Color information is very important in human perception. Therefore it has attracted the interest of the researchers working in computer vision and image processing quite early, i.e. at the beginning of the seventies. However, color image processing has not experienced the same fast development

with other areas in digital image processing. There are several reasons for this delay. First of all, the development of monochromatic image processing techniques is the first step towards color image processing. Therefore scientists concentrated their efforts on black and white (BW) image processing. The second reason is that no accurate models of the human color perception were available, thus making the development of color image processing algorithms difficult. The third reason is the large amount of data and computation required (almost 1 Mbyte for a 512×512 RGB image).

Early applications of color are in false color, pseudocolor [1], color composition and ratioing [2], [3]. False color and pseudocolor are color mappings to enhance detectability of certain objects in the image by a human observer. In pseudocolor, the original image is not a color image. Color composition and ratioing are techniques commonly applied to multispectral imagery to reveal subtle variations that exist between the individual spectral components.

Despite those difficulties in the relatively new and still largely unexplored area of color image processing, several efforts have been made. The relevant research can be divided into the following areas:

1.1 Color Vision Modelling

A lot of research has been done by researchers to model the early stages of the human chromatic perception system. Some of the early results are summarized in classical books [1], [2], [4]. More recent results on color image formation can be found in [5–7]. Problems as distinguishing highlights, shadows and material changes, as well as chromatic constancy under different illuminants are discussed in these references.

The trichromatic nature of light is represented by three component color spaces. Thus, a color image requires three times the data and storage of its monochrome equivalent. As such, efficient compression schemes should be developed for color images. Color image compression is a relatively active research area. In [8] a new adaptive vector quantization scheme with codebook replenishment for color image sequence compression is presented. In [9], an *interpolative vector quantizer* (IVQ) is invented to reduce the block effect of coded color pictures. A generalized variable-stage motion search motion compensated algorithm is given in [10]. Considerable savings in computations and additional compression are achieved in the algorithm without compromising the picture quality. In [11], RGB (red, green, blue) images are first transformed into YIQ coordinates. Color signals are multichannel (vector) signals. However, no effort has been made to develop a theory for multi-channel color processing although such efforts have already

been made in other areas, e.g. geophysics [12]. The development of such a theory will be beneficial for all other problems in color image processing.

1.2 Color Image Restoration and Enhancement

The brute force approach to color image restoration consists in performing three restorations in the R, G and B channel of a color image. However, it is known that the R, G and B components of a color image are generally correlated. Therefore, the use of this correlation in the restoration process is of great advantage. Karhunen-Loeve (K-L) transformation is used for the decorrelation of the three image components and then Wiener filtering is applied to the decorrelated components [13]. Also RGB to XYZ and YIQ transformations have been also proposed for decorrelation purposes. Both methods produce correlated noise components in each channel. Blur identification from blurred color images, as well as constrained least squares restoration are presented in [14]. The modelling of each of the R,G,B components has been described in [15] for image restoration purposes. The fact that the saturation component Y in the YIQ representation carries much of the high frequency image components has been used for color image enhancement by processing the saturation component [16]. Finally, a modification of the median filters has been proposed in [17] for color image filtering.

Multiple restorations of monochromatic images do not deblur a color image. Optimal restoration of a color image has been found in decorrelated component restoration [18]. To obtain truly decorrelated components, the tedious K-L transformation must be applied. Since YIQ transformation approximates the K-L transformation, it can be used instead. Research has found that deblurring only the Y component of a color image is often sufficient. In [18], deblurring only the Y component using the theoretical Wiener filter is suggested. [15] presents the use of reduced updated Kalman filter on Y.

Early color image enhancement techniques concentrated in satellite imagery [3]. In [16] the sharpening of color images in video and photography using the LHS (luminance, hue, saturation) color space is considered. For outdoor scenes, saturation data contain more high frequency energy than the luminance data. The algorithm adds spatial frequencies from saturation data to luminance data in order to enhance edges in the luminance component. The same approach can be applied to adaptive sharpness enhancement and adaptive contrast enhancement.

The modelling of color vision is critical in enhancement of color images. Early models are reported in [1]. More recent research considering

physical factors such as lighting and reflectance are highlighted in [19] and [6]. In [20], experimental results show that much larger color differences are required for the recognition of small irregular objects on a CRT than those assumed from the conventional color discrimination data for detection or color matching.

Image processing of monochromatic images is very rich in both linear and nonlinear techniques for image restoration, filtering and enhancement. The extension of those methods to color image processing is not trivial if the correlation of the color channels is taken into account. Therefore the development of techniques such as Wiener filtering, minimum entropy filtering is an open problem. Furthermore, the development of nonlinear color image filters (e.g. homomorphic filters, order statistics filters, morphological filters) is an open problem, too.

1.3 Color Edge Detection and Image Segmentation

Color edge detection has been scarcely discussed in the literature, although it is known that color carries edge information. Edges are assumed to be intensity discontinuities in the three components, which are relatively independent with the constraint of having the same orientation [21]. The achromatic Hueckel edge detector is extended to color edges. Luminance edges are found to contain most of the information required to obtain object boundaries. However, for images of poor contrast or poor illumination, edges are present only in the chromatic components. Also the use of different color coordinates for edge detection has been investigated in [22]. More work has been devoted to color image segmentation, where color information has been recognized as important. RGB, YIQ and HSI (hue saturation intensity) components have been used for color image segmentation [23–26].

An important aspect of color image segmentation is the choice of suitable color features. RGB, YIQ, HSI decompositions have been used in various segmentation research. [23] suggests using the Karhunen-Loeve (K-L) transformation as test, and after running it on 8 different images,

$$I_1 = (R+G+B)/3\ , \quad I_2 = R-B\ , \quad I_3 = (2G-R-B)/2$$

are found to be the set of effective color features. This set may be useful in other related areas, e.g. in edge extraction. The use of the XYZ color space and of the three dimensional histograms for segmentation was investigated in [25]. Also the Munsell color system and color histograms were applied for segmentation purposes in [27]. Finally, multichannel AR models were used for color texture description and region segmentation in [28]. In recent years, there has been a burst in the literature about edge detectors. Color

edge detection, when carefully defined, can take great advantage of all these recent developments.

1.4 Color Image Coding

Color image coding has been one of the most active research areas in the recent years. An excellent review of the early research can be found in [29]. In more recent approaches motion compensated color image coding was considering in [10]. In most cases in color image coding color transformations were used to reduce the channel correlation [29], [11].

1.5 Digital and High Definition TV

The main advances in color image processing are expected to come from the areas of digital color TV [30], [31] and high definition TV (HDTV) [32]. A lot of technological research and development is conducted in those areas mainly in the private sector. The commercial application of both systems is expected to produce new vistas in the theoretical and applied research in color images processing.

1.6 Fast Implementation Techniques for Color Image Processing

The processing of color images requires fast processing of huge amounts of data. In addition, many applications in telecommunications, moving object recognition and tracking, traffic monitoring (of planes, rockets, satellites, cars and fishes), automatic industrial control and inspection [33], remote sensing and robotic vision [34] require real-time processing. Thus, the acquisition, processing and display of images must be executed in fractions of a second.

For the processing of color images, multivariable 3-input, 3-output 2-D digital filters are used, considering decomposition in the R, G and B components. Assuming that the three image components are decorrelated (after the application of the Karhunen-Loeve transformation), three independent single-input, single-output (SISO) 2-D digital filters are needed for the processing of each monochromatic image. Additional processing is required for the correlated noise components in each channel.

2-D digital filters are implemented both in software and hardware. However, the requirement of very fast processing dictates the use of special purpose hardware implementations. VLSI technology has recently resulted in enormous hardware facilities at very low costs for the implementation of

high complexity algorithms, and leads to a reconsideration of design and implementation criteria in digital signal processing. Specifically, past approaches based on the minimization of dynamic elements and of the quantization noise are replaced by a new set of criteria such as concurrency, parallelism, pipelining, modularity, flexibility, throughput rate and reduction of the latency of the number of shared buses and of the communication paths.

The *VLSI array processors* (APs), which are special purpose, locally interconnected computing networks [35]–[38], ideally suit the fast implementation of 1-D and 2-D digital filters [39]–[44]. The APs maximize concurrency by exploiting both parallelism and pipelining. Prominent classes of VLSI APs are systolic and wavefront arrays. A systolic array is a network of elementary processor elements (PEs) that rhythmically compute and pass the data through the system. A wavefront array may be regarded as a systolic array with local synchronization using data-flow control.

The underlying realization structures of the 2-D digital filters that will process 2-D color images will exploit the ideas of *matrix decomposition and block processing*. In the block processing structures a number of input pixels are processed simultaneously. Thus, high sampling rates are permitted, since the operations in the parallel branches are executed individually and simultaneously on a common input array [45]–[48]. In general, if L pixels are processed simultaneously, then the throughput rate is increased by L. L and the throughput rate are confined only by practical limitations such as hardware complexity and input-output bound.

Matrix decomposition approaches have been used for the modular, parallel implementation of linear and quadratic nonlinear 2-D digital filters that are appropriate for processing monochromatic images [49], [50]. In this technique, the coefficient matrices are decomposed in a product of other matrices; finally, the whole filter is composed of combinations of parallel branches, each one consisting of 1-D terms. The techniques of block processing and matrix decomposition have been combined in order to make use of all their advantages [51], [52].

This paper presents fast implementation architectures of 3-input, 3-output 2-D linear digital filters for color image processing that are based on matrix decompositions. The concept of block processing may be also exploited in order to increase concurrency and achieve real-time color image processing. The resulting structures are modular, regular, have high inherent parallelism and are easily pipelined, so that they may be implemented via VLSI array processors. The proposed technique may be applied for the fast and efficient implementation of 3-input, 3-output 2-D nonlinear Volterra digital filters.

2. Modelling of Multichannel 2-D Linear Digital Filters

Since the color images have three independent coordinates, the R, G and B images, the associated 2-D shift-invariant multivariable linear filter with three inputs and three outputs, is described by the matrix convolution equation:

$$Y(k,l) = \sum_{i=0}^{n_1}\sum_{j=0}^{m_1} A_{ij}U(k-i,l-j) - \sum_{\substack{i=0\\(i,j)\neq(0,0)}}^{n_2}\sum_{j=0}^{m_2} B_{ij}Y(k-i,l-j)\ , \quad (2.1)$$

where

$$U(k,l) = \begin{bmatrix} u_R(k,l) \\ u_G(k,l) \\ u_B(k,l) \end{bmatrix}, \quad Y(k,l) = \begin{bmatrix} y_R(k,l) \\ y_G(k,l) \\ y_B(k,l) \end{bmatrix} \quad (2.2)$$

are the (k,l)th input and output vectors, respectively.

The application of the 2-D z-transform to the matrix equation (2.1) gives:

$$Y(z_1,z_2) = H(z_1,z_2)U(z_1,z_2)\ , \quad (2.3)$$

where $H(z_1,z_2)$ is the 3×3 transfer function matrix of the 2-D shift-invariant multivariable linear filter, having the form

$$H(z_1,z_2) = \begin{bmatrix} h_{RR}(z_1,z_2) & h_{RG}(z_1,z_2) & h_{RB}(z_1,z_2) \\ h_{GR}(z_1,z_2) & h_{GG}(z_1,z_2) & h_{GB}(z_1,z_2) \\ h_{BR}(z_1,z_2) & h_{BG}(z_1,z_2) & h_{BB}(z_1,z_2) \end{bmatrix}. \quad (2.4)$$

The diagonal transfer functions elements represent the SISO 2-D digital filters that process the monochromatic R, G and B images, while the off-diagonal transfer functions stand for the input-output relations of correlated noise components among the monochromatic channels. Specifically, the $h_{ij}(z_1,z_2)$ term marks the transfer function relating the ith output $Y_i(z_1,z_2)$ with the jth input $U_j(z_1,z_2)$ and is given by

$$h_{ij}(z_1,z_2) = \frac{Y_i(z_1,z_2)}{U_j(z_1,z_2)} = \frac{a_{ij}(z_1,z_2)}{b_{ij}(z_1,z_2)}, \quad i,j = R,G,B\ , \quad (2.5)$$

where

$$a_{ij}(z_1,z_2) = \sum_{s=0}^{n_1}\sum_{t=0}^{m_1} a_{st}^{ij} z_1^s z_2^t\ , \quad (2.6)$$

$$b_{ij}(z_1,z_2) = 1 + \sum_{\substack{s=0\\(s,t)\neq(0,0)}}^{n_2}\sum_{t=0}^{m_2} b_{st}^{ij} z_1^s z_2^t \quad (2.7)$$

and $U_j(z_1,z_2)$, $Y_i(z_1,z_2)$ denote the 2-D z-transforms of $u_j(k,l)$ and $y_i(k,l)$, respectively.

3. Matrix Decomposition Structures of 2-D Multivariable Linear Digital Filters

The decomposition matrix approach, as well as the block processing approach, should be applied to the 3-input, 3-output 2-D digital filter with the transfer function matrix $H(z_1, z_2)$. In the sequel, this will be done independently for the cases of FIR and IIR filters.

FIR Filters

For the case of FIR filters, (2.1) is reduced to the nonrecursive matrix equation

$$Y(k,l) = \sum_{i=0}^{n_1} \sum_{j=0}^{m_1} A_{ij} U(k-i, l-j) \,. \tag{3.1}$$

The application of the 2-D z-transform to (3.1) gives:

$$Y(z_1, z_2) = A(z_1, z_2) U(z_1, z_2) \tag{3.2}$$

and $U(z_1, z_2)$, $Y(z_1, z_2)$ denote the 2-D z-transforms of $U(k,l)$ and $Y(k,l)$, respectively. It can be seen from (3.2) that the 3x3 polynomial matrix $A(z_1, z_2)$ is the transfer function matrix $H(z_1, z_2)$ of the multivariable 2-D FIR filter and has the form

$$H(z_1, z_2) = A(z_1, z_2) = \sum_{i=0}^{n_1} \sum_{j=0}^{m_1} A_{ij} z_1^i z_2^j \,, \tag{3.3}$$

where

$$A_{ij} = \begin{bmatrix} a_{ij}^{11} & a_{ij}^{12} & a_{ij}^{13} \\ a_{ij}^{21} & a_{ij}^{22} & a_{ij}^{23} \\ a_{ij}^{31} & a_{ij}^{32} & a_{ij}^{33} \end{bmatrix} \tag{3.4}$$

are 3x3 constant coefficient matrices.

The polynomial matrix $A(z_1, z_2)$ may be written in the form

$$A(z_1, z_2) = \begin{bmatrix} Z_1^T \hat{A}_{11} z_2 & Z_1^T \hat{A}_{12} Z_2 & Z_1 \hat{A}_{13} Z_2 \\ Z_1^T \hat{A}_{21} z_2 & Z_1^T \hat{A}_{22} Z_2 & Z_1 \hat{A}_{23} Z_2 \\ Z_1^T \hat{A}_{31} z_2 & Z_1^T \hat{A}_{32} Z_2 & Z_1 \hat{A}_{33} Z_2 \end{bmatrix} \,, \tag{3.5}$$

where

$$Z_1 = \begin{bmatrix} 1 \\ z_1 \\ \vdots \\ z_1^{n_1} \end{bmatrix} \,, \quad Z_2 = \begin{bmatrix} 1 \\ z_2 \\ \vdots \\ z_2^{m_1} \end{bmatrix} \,, \tag{3.6}$$

and $\hat{A}_{kl} \in \mathbf{R}^{(n_1+1)(m_1+1)}$ are coefficient matrices defined by

$$\hat{A}_{kl} = \begin{bmatrix} a_{00}^{kl} & a_{01}^{kl} & \dots & a_{0,m_1}^{kl} \\ a_{10}^{kl} & a_{11}^{kl} & \dots & a_{1,m_1}^{kl} \\ \vdots & \vdots & & \vdots \\ a_{n_1,0}^{kl} & a_{n_1,1}^{kl} & \dots & a_{n_1,m_1}^{kl} \end{bmatrix} . \tag{3.7}$$

Equivalently, (3.5) may be rewritten in the concise form

$$A(z_1, z_2) = [I_3 \otimes Z_1^T]\hat{A}[I_3 \otimes Z_2] , \tag{3.8}$$

where $\otimes$ denotes the Kronecker product. The Kronecker product of two matrices $X \in \mathbf{R}^{n \times m}$ is defined in [53] as

$$X \otimes Y = \begin{bmatrix} x_{11}Y & x_{12}Y & \dots & x_{1m}Y \\ x_{21}Y & x_{22}Y & \dots & x_{2m}Y \\ \vdots & \vdots & & \vdots \\ x_{n1}Y & x_{n2}Y & \dots & x_{nm}Y \end{bmatrix} .$$

The constant coefficient matrix $\hat{A} \in \mathbf{R}^{3(n_1+1)\times 3(m_1+1)}$ in (3.8) is given by

$$\hat{A} = \begin{bmatrix} \hat{A}_{11} & \hat{A}_{12} & \hat{A}_{13} \\ \hat{A}_{21} & \hat{A}_{22} & \hat{A}_{23} \\ \hat{A}_{31} & \hat{A}_{32} & \hat{A}_{33} \end{bmatrix} . \tag{3.9}$$

Alternatively, the polynomial matrix $A(z_1, z_2)$ may be written in the form

$$A(z_1, z_2) = [Z_1^T \otimes I_3]A[Z_1 \otimes I_3] , \tag{3.10}$$

where the constant coefficient matrix $A \in \mathbf{R}^{3(n_1+1)\times 3(m_1+1)}$ in (3.10) is given by

$$A = \begin{bmatrix} A_{00} & A_{01} & \dots & A_{0,m_1} \\ A_{10} & A_{11} & \dots & A_{1,m_1} \\ \vdots & \vdots & & \vdots \\ A_{n_1,0} & A_{n_1,1} & \dots & A_{n_1,m_1} \end{bmatrix} , \tag{3.11}$$

where A_{ij}, $i = 0, 1, 2, ..., n_1$, $j = 0, 1, 2, ..., m_1$ are the constant coefficient matrices given in (3.4). Note that A contains the same elements with those of $\hat{A}$, reorganized in a different way.

In the sequel the matrix decomposition approach will be applied on the matrix A (equivalently, the decomposition may be applied on $\hat{A}$) as follows:

$$A = RS , \tag{3.12}$$

where $R \in \mathbf{R}^{3(n_1+1)\times p}$ and $S \in \mathbf{R}^{p\times 3(m_1+1)}$ are constant matrices. The decomposition of A in (3.10) may be achieved in a number of ways so that the Sylvester's inequality

$$\operatorname{rank}(R) + \operatorname{rank}(S) - p \leq \operatorname{rank}(A) \leq \min\{\operatorname{rank}(R), \operatorname{rank}(S)\} \tag{3.13}$$

holds.

One possible decomposition is achieved by selecting R as a $3(n_1+1)\times 3(n_1+1)$ square nonsingular matrix and then calculating S by $S = R^{-1}A$. In this case, it is desirable for the matrix A to have a known inverse matrix, in order to reduce the calculations for the determination of S.

Substituting (3.12) in (3.10), we obtain:

$$\begin{aligned} A(z_1, z_2) &= [(Z_1^T \otimes I_3)R][S(Z_2 \otimes I_3)] \\ &= \overline{R}(z_1)\overline{S}(z_2) \\ &= [(r_1(z_1)]^T\ [r_2(z_1)]^T\ [r_3(z_1)]^T]^T\ [s^1(z_2)\ s^2(z_2)\ s^3(z_2)] \\ &= \begin{bmatrix} r_1(z_1)s^1(z_2) & r_1(z_1)s^2(z_2) & r_1(z_1)s^3(z_2) \\ r_2(z_1)s^1(z_2) & r_2(z_1)s^2(z_2) & r_2(z_1)s^3(z_2) \\ r_3(z_1)s^1(z_2) & r_3(z_1)s^2(z_2) & r_3(z_1)s^3(z_2) \end{bmatrix}, \end{aligned} \tag{3.14}$$

where $r_i(z_1)$, $i = 1, 2, 3$ and $s^j(z_2)$, $j = 1, 2, 3$, are the ith row and jth column of the matrices $\overline{R}(z_1)$ and $\overline{S}(z_2)$, respectively.

Substituting (3.14) in (3.2), we obtain that the τth, $\tau = 1, 2, 3$, partial output $Y_T(z_1, z_2)$ is given by (*Fig. 1*)

$$\begin{aligned} Y_\tau(z_1, z_2) &= \sum_{i=1}^{3} Y_{\tau i}(z_1, z_2) \\ &= \sum_{i=1}^{3} [r_\tau(z_1)s^i(z_2)]U_i(z_1, z_2) \\ &= \sum_{i=1}^{3} \left[\sum_{j=1}^{p} r_{\tau,j}(z_1)s_{ji}(z_2)\right] U_i(z_1, z_2), \quad \tau = 1, 2, 3, \end{aligned} \tag{3.15}$$

where $Y_{Ti}(z_1, z_2)$ represents the τth, $\tau = 1, 2, 3$, partial output, which is influenced by the ith input $U_i(z_1, z_2)$, and $r_{ij}(z_1)$, $s_{ij}(z_2)$ denote the (i, j)th elements of the matrices $\overline{R}(z_1)$ and $\overline{S}(z_2)$, respectively.

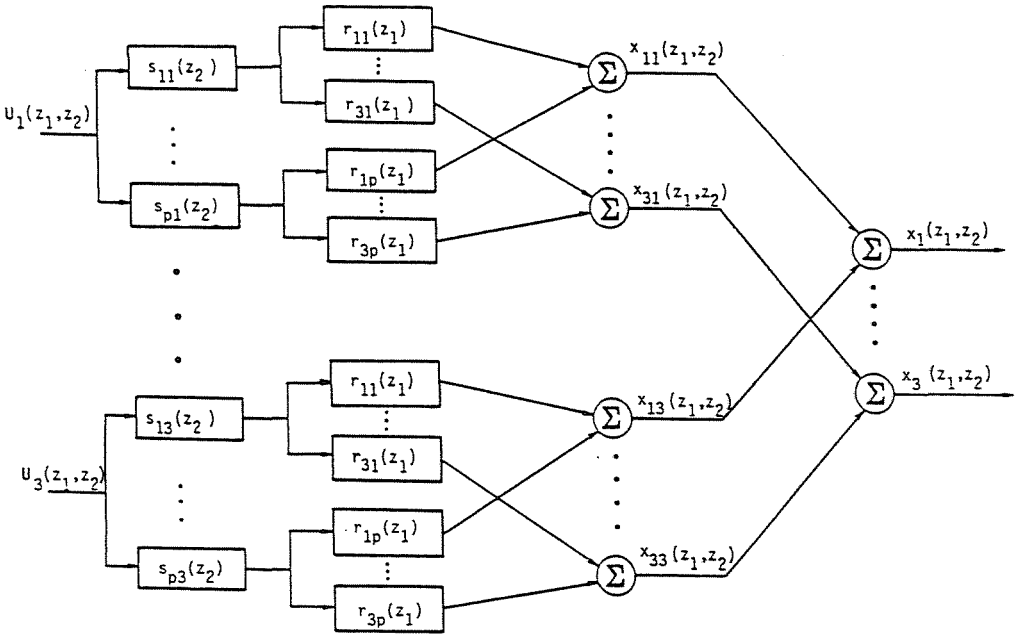

Fig. 1. The matrix decomposition-based realization of the 2-D FIR filter with transfer function matrix $H(z_1, z_2) = A(z_1, z_2)$, according to (3.15)

IIR Filters

The application of the 2-D z-transform to (2.1) provides the 3×3 transfer function matrix of the multivariable 2-D IIR filter

$$H(z_1, z_2) = [I_3 + B(z_1, z_2)]^{-1} A(z_1, z_2) \,, \tag{3.16}$$

where $A(z_1, z_2)$ is given by (3.3) and $B(z_1, z_2)$ by

$$B(z_1, z_2) = \sum_{\substack{i=0 \\ (i,j)\neq(0,0)}}^{n_2} \sum_{j=0}^{m_2} B_{ij} z_1^i z_2^j \,. \tag{3.17}$$

The IIR filter described by the transfer function matrix (3.16) may be implemented in two alternative forms, direct form I and II, as a cascade configuration of two 3-input, 3-output filters. Specifically, the direct form I

consists of the 2-D FIR filter with a transfer function matrix $A(z_1, z_2)$, in cascade with an IIR filter with a transfer function matrix $[I_3 + B(z_1, z_2)]^{-1}$ (*Fig. 2*). The direct form II is obtained if the order of the 2-D FIR and IIR filters is reversed (*Fig. 3*). Since both filters are space-invariant, their order can be reversed without altering the final transfer function matrix $H(z_1, z_2)$.

Direct Form I Decomposition

The transfer function matrix $A(z_1, z_2)$ of the 2-D FIR filter is written in the form (3.14), according to the matrix decomposition (3.12). The output $X(z_1, z_2)$ of the FIR filter is given by (3.15).

The all-pole 2-D IIR filter, described by the transfer function matrix $[I_3 + B(z_1, z_2)]^{-1}$, is implemented with unities in the forward branch and the polynomial matrix $B(z_1, z_2)$ in the feedback branch (*Fig. 2*), which is decomposed similarly to $A(z_1, z_2)$ in (3.14). In fact, $B(z_1, z_2)$ may be written as

$$B(z_1, z_2) = [\overline{Z}_1^T \otimes I_3] B [\overline{Z}_2 \otimes I_3] , \tag{3.18}$$

where the constant coefficient matrix $B \in \mathbf{R}^{3(n_2+1)\times 3(m_2+1)}$ in (3.18) is given by

$$B = \begin{bmatrix} 0 & B_{01} & \dots & B_{0,m_2} \\ B_{10} & B_{11} & \dots & B_{1,m_2} \\ \vdots & \vdots & & \vdots \\ B_{n_2,0} & B_{n_2,1} & \dots & B_{n_2,m_2} \end{bmatrix} \tag{3.19}$$

and

$$\overline{Z}_1 = \begin{bmatrix} 1 \\ z_1 \\ \vdots \\ z_1^{n_2} \end{bmatrix} , \quad \overline{Z}_2 = \begin{bmatrix} 1 \\ z_2 \\ \vdots \\ z_2^{m_2} \end{bmatrix} . \tag{3.20}$$

It can be seen from (3.18) and (3.19) that $B(z_1, z_2)$ does not contain a constant term. This fact ensures the fundamental necessary realizability condition that the corresponding implementation structure does not contain any delay-free loops. It is emphasized that a structure is realizable if and only if none of the parallel branches has a constant term. This latter necessary and sufficient condition can be always satisfied by the decomposition structure, by selecting properly the auxiliary matrices Q and S in the decomposition of the matrix B

$$B = QS . \tag{3.21}$$

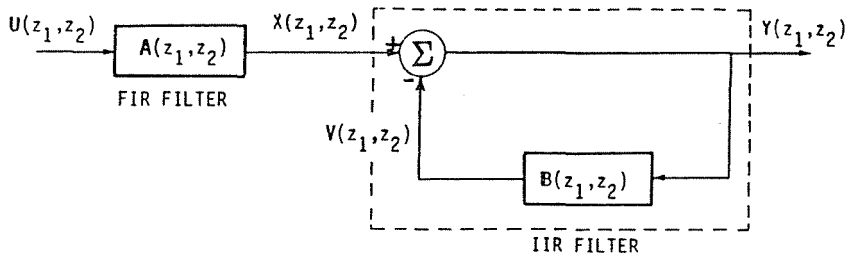

Fig. 2. Block diagram for direct form I realization of the 2-D IIR filter with transfer function matrix $H(z_1, z_2) = [I_3 + B(z_1, z_2)]^{-1} A(z_1, z_2)$

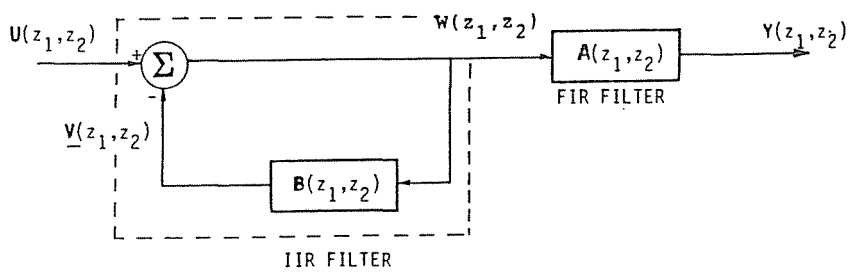

Fig. 3. Block diagram for direct form II realization of the 2-D IIR filter with transfer function matrix $H(z_1, z_2) = [I_3 + B(z_1, z_2)]^{-1} A(z_1, z_2)$

A matrix form of S ensuring realizability is the nonsingular $3(m_2 + 1) \times 3(m_2 + 1)$ matrix

$$S = \begin{bmatrix} S_{00} & S_{01} & \dots & S_{0,m_2} \\ 0 & S_{11} & \dots & S_{1,m_2} \\ \vdots & \vdots & & \vdots \\ 0 & S_{m_2,1} & \dots & S_{m_2,m_2} \end{bmatrix}, \tag{3.22}$$

where S_{ij} , $i,j = 0,1,...,m_2$ are 3×3 constant matrices. Indeed, in that case, the matrix Q has the form

$$Q = \begin{bmatrix} 0 & Q_{01} & \dots & Q_{0,m_2} \\ Q_{10} & Q_{11} & \dots & Q_{1,m_2} \\ \vdots & \vdots & & \vdots \\ Q_{n_2,0} & Q_{n_2,1} & \dots & Q_{n_2,m_2} \end{bmatrix} , \tag{3.23}$$

where Q_{ij}, $i = 0,1,...,n_2$, $j = 0,1,...,m_2$ are 3×3 constant matrices. In correspondence to (3.14), the polynomial matrix $B(z_1,z_2)$ may be written in the form

$$\begin{aligned} B(z_1,z_2) &= [(Z_1^T \otimes I_3)Q][S(Z_2 \otimes I_3)] \\ &= \overline{Q}(z_1)\overline{S}(z_2) \\ &= [(q_1(z_1)]^T\ [q_2(z_1)]^T\ [q_3(z_1)]^T]^T[s^1(z_2)\ s^2(z_2)\ s^3(z_2)] \\ &= \begin{bmatrix} q_1(z_1)s^1(z_2) & q_1(z_1)s^2(z_2) & q_1(z_1)s^3(z_2) \\ q_2(z_1)s^1(z_2) & q_2(z_1)s^2(z_2) & q_2(z_1)s^3(z_2) \\ q_3(z_1)s^1(z_2) & q_3(z_1)s^2(z_2) & q_3(z_1)s^3(z_2) \end{bmatrix} , \end{aligned} \tag{3.24}$$

where $q_i(z_1)$, $i = 1,2,3$ and $s^j(z_2)$, $j = 1,2,3$, are the ith row and jth column of the matrices $\overline{Q}(z_1)$ and $\overline{S}(z_2)$, respectively. If the matrices have the form of (3.22) and (3.23), it can be easily seen that none of the elements $q_i(z_1)s^j(z_2)$, $i = 1,2,3$, $j = 1,2,3$ in the decomposed implementation has a constant term. This fact ensures that there are not any delay-free loops in the feedback branches.

The matrix S in (3.21) is generally different from the matrix S in (3.12). For the sake of modularity, the matrix S may be selected to be the same in the matrix decompositions (3.12) and (3.21), provided that $m_1 = m_2$.

The feedback branches in *Fig. 2* are described by the vector equation

$$V(z_1,z_2) = B(z_1,z_2)Y(z_1,z_2). \tag{3.25}$$

The substitution of (3.24) in (3.25) results in that the τth, =1,2,3, partial feedback signal is given by (*Fig. 4*)

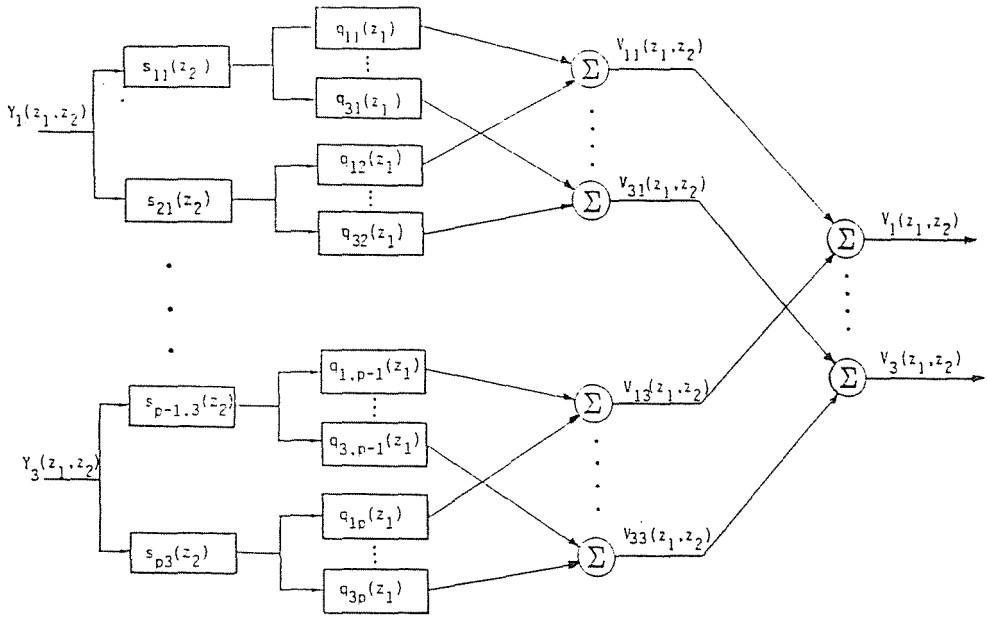

Fig. 4. The direct form I matrix decomposition-based realization of the 2-D IIR filter

$$
\begin{aligned}
V_\tau(z_1,z_2) &= \sum_{i=1}^{3} V_{\tau i}(z_1,z_2) \\
&= \sum_{i=1}^{3} [q_1(z_1)s^i(z_2)]Y_i(z_1,z_2) \\
&= \sum_{i=1}^{3} \left[\sum_{j=1}^{p} q_{\tau j}(z_1)s_{ji}(z_2)\right] U_i(z_1,z_2)\ , \quad \tau = 1,2,3\ , \quad (3.26)
\end{aligned}
$$

where $V_{\tau i}(z_1,z_2)$ represents the τth, $\tau = 1,2,3$, feedback signal, which is influenced by the ith output $Y_i(z_1,z_2)$. In (3.26) it has been considered that p is now the number of columns of Q (note that p denotes the number of columns of R in (3.15)).

Direct Form II Decomposition

The direct form II decomposition of the 2-D IIR filter with transfer function $H(z_1, z_2)$ is described by the equations (*Fig. 3*):

$$W(z_1, z_2) = [I_3 + B(z_1, z_2)]^{-1} U(z_1, z_2) \tag{3.27}$$

and

$$Y(z_1, z_2) = A(z_1, z_2) W(z_1, z_2) \ . \tag{3.28}$$

Moreover, the feedback branches in *Fig. 3* are described by the vector equation

$$V(z_1, z_2) = B(z_1, z_2) W(z_1, z_2) \ . \tag{3.29}$$

The matrix convolution equations corresponding to (3.27) and (3.28) in the space domain are

$$W(k,l) = U(k,l) - \sum_{\substack{i=0 \\ (i,j)\neq(0,0)}}^{n_2} \sum_{j=0}^{m_2} B_{ij} W(k-i, l-j) \ , \tag{3.30}$$

$$Y(k,l) = \sum_{i=0}^{n_1} \sum_{j=0}^{m_1} A_{ij} W(k-i, l-j) \ . \tag{3.31}$$

The substitution of the polynomial matrices $B(z_1, z_2)$ and $A(z_1, z_2)$ in their decomposed forms (3.24) and (3.14) in (3.28) and (3.29), respectively, gives

$$\begin{aligned} Y_\tau(z_1, z_2) &= \sum_{i=1}^{3} Y_{\tau i}(z_1, z_2) \\ &= \sum_{i=1}^{3} [r_\tau(z_1) s^i(z_2)] W_i(z_1, z_2) \ , \quad \tau = 1, 2, 3 \ , \end{aligned} \tag{3.32}$$

$$\begin{aligned} V_\tau(z_1, z_2) &= \sum_{i=1}^{3} V_{\tau i}(z_1, z_1) \\ &= \sum_{i=1}^{3} [q_\tau(z_1) s^i(z_2)] W_i(z_1, z_2) \ , \quad \tau = 1, 2, 3 \ . \end{aligned} \tag{3.33}$$

where $Y_{\tau i}(z_1, z_2)$ and $V_{\tau i}(z_1, z_2)$ represent the τth, $\tau = 1, 2, 3$ partial output of the feedback signal, respectively, which are influenced by the ith intermediate signal $W_i(z_1, z_2)$.

Comparing (3.32) and (3.33), it can be seen that the 1-D factors $s^i(z_2)$ (if they have been selected to be the same in both the decompositions), may be shared by the IIR all-pole filter (3.27) and the FIR filter (3.28). Sharing $s^i(z_2)$ among the feedback branches of the IIR filter and the forward branches of the FIR filter results in a reduction of the hardware and storage requirements (*Fig. 5*).

4. Matrix Decomposition Structures of 2-D Multivariable Volterra Nonlinear Digital Filters

In a viable approach to the description of nonlinear 1-D and 2-D digital filters that are suitable to analytic characterization, analysis and synthesis are based upon the truncated Volterra series [54], [55]. This approach satisfies the desire to obtain characterization procedures for nonlinear systems that retain a part of simplicity and analytic representation, since the filter's output is linear in respect of the Volterra kernels. Implementations of 1-D and 2-D quadratic nonlinear digital filters based on the Volterra series description [55], as well as highly nonlinear digital filters implementation [56] have been already presented. Efficient implementation structures, based on appropriate matrix decompositions, of multi-input, multi-output 2-D nonlinear Volterra digital filters may be derived by applying the presented multichannel technique to the methodology developed for single-input, single output Volterra digital filters [55], [56].

5. Concluding Remarks

Fast implementation architectures of 3-input, 3-output 2-D linear digital filters for color image processing that are based on matrix decompositions have been presented. The proposed implementation structures are characterized by a high degree of concurrency (due to combined inherent parallelism and data pipelining), modularity and regularity. They are ideally suited for implementation using systolic and wavefront array processors, as well as by other classes of VLSI APs such as the MIMD (multiple instructions, multiple data) and the SIMD (single instruction, multiple data). Special Walsh-Hadamard transforms may be used for the involved matrix decompositions in order to avoid intermediate calculations [52]. The block processing of color images using the matrix decomposition approach is also possible.

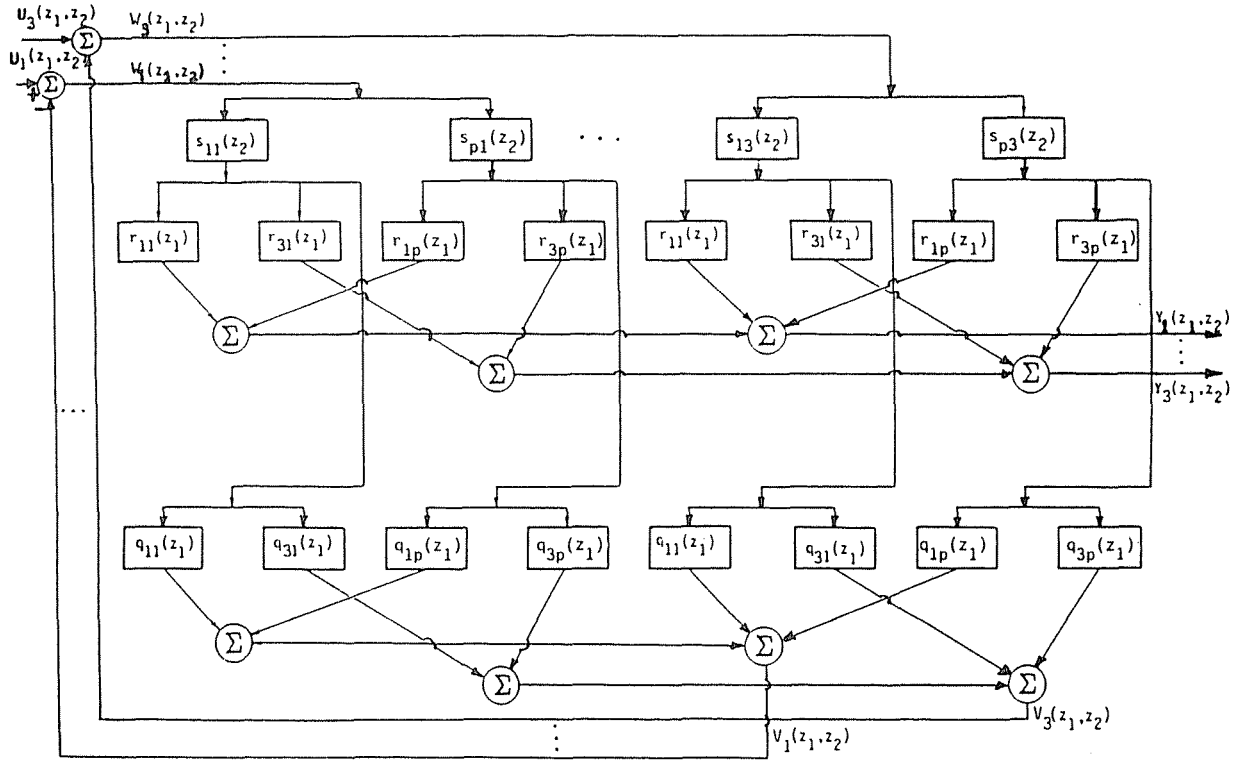

Fig. 5. The direct form II matrix decomposition-based realization of the 2-D IIR filter, where the factors $s^i(z_2)$ are shared among the feedback and the forward branches.

References

1. PRATT, W. K.: Digital Image Processing, Prentice-Hall, 1978.
2. GREEN, W. B.: Digital Image Processing. A Systems Approach, Van Nostrand Reinhold, 2nd edition, 1988.
3. MATHER, P. M.: The Use of Color in Enhancing Satellite Imagery of the Earth, *Proceedings of the International Conference on Color in Information Technology and Visual Displays, IERE Publication*, No. 61, pp. 55–58, March 1988.
4. LEVINE, M. D.: Vision in Man and Machine, McGraw-Hill, 1985.
5. GERSHON, R.: The Use of Color in Human Vision, Ph.D. Thesis, University of Toronto, 1987.
6. WANDELL, B. A.: The Synthesis and Analysis of Color Images, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. PAMI–9, No. 1, pp. 2–13, 1987.
7. BUMBACA, F. – SMITH, K. C.: Design and Implementation of a Color Vision Model for Computer Applications, *Computer Vision Graphics and Image Processing*, Vol. 39, pp. 226–245, 1987.
8. YEH, C. L.: Color Image Sequence Compression Using Adaptive Binary Tree Vector Quantization with Codebook Replenishment, *Proceedings of the IEEE 1987 International Conference on Acoust., Speech and Signal Processing*, Vol. 2, pp. 1059–1062, 1987.
9. HANG – HASKELL: Interpolative Vector Quantization of Color Images, *IEEE Trans. on Communications*, Vol. COM–36, No. 4, pp. 465–470, April 1988.
10. KWATRA, S. – LIN, C. – WHYTE, W.: An Adaptive Algorithm for Motion Compensated Color Image Coding, *IEEE Trans. on Communications*, Vol. COM–35, pp. 747–754, July 1987.
11. GHARAVI, H.: Sub-band Coding of Monochrome and Color Images, *IEEE Trans. on Circuits and Systems*, Vol. CAS–35, No. 2, pp. 207–214, Febr. 1988.
12. TREITEL, S.: Principles of Digital Multichannel Filtering, *Geophysics*, Vol. 35, No. 5, pp. 785–811, Oct. 1970.
13. HUNT, B. R. – KUEBLER, O.: Karhunen-Loeve Multispectral Image Restoration, Part I: Theory, *IEEE Transactions on Acoustics, Speech and Signal Processing*, Vol. ASSP–32, No. 3, pp. 592–599, June 1984.
14. WESTERINK, P. H. – BIEMOND, J. – DE BRUIN, P. H. L.: Digital Color Image Restoration, in Signal Processing III: Theories and Applications, I. T. Young et al. editors, Elsevier, 1986.
15. ANGUIN, D. L. – KAUFMAN, H.: Effects of Modelling Domains on Recursive Color Image Segmentation, *Proceedings of the IEEE International Conference on Acoust., Speech and Signal Processing*, pp. 1229–1231, 1987.
16. STRICKLAND, R. – MCDONNELL, C. K. W.: Digital Color Image Enhancement Based on the Saturation Component, *SPIE Optical Engineering*, Vol. 26, No. 7, pp. 609–616, July 1987.
17. ASTOLA, J. – HAAVISTO, P. – HEINONEN, P. – NEUVO, Y.: Median Type Filters for Color Signals, *Procedings of the 1988 IEEE International Symposium on Circuits and Systems*, pp. 1753–1756, 1988.
18. HUNT, B. R. – OLAF, K.: The Theory of Optimal Multi-Spectral Image Restoration, *Proceedings of SPIE Applications of Digital Image Processing*, Vol. 397.
19. GERSHON, R.: Aspects of Perception and Computation in Color Vision, *Computer Vision, Graphics and Image Processing*, Vol. 32, No. 2, pp. 244–247, Nov. 1985.

20. PHILLIPS, P. L.: Minimum Color Differences Required to Recognise Small Objects on a Color CRT, *International Conference on Color in Information Technology and Visual Displays, IERE Publication*, No.61.
21. NEVATIA, R.: A Color Edge Detector and its Use in Scene Segmentation, *IEEE Transactions on Systems, Man and Cybernetics*, Vol. SMC-7, No. 11, pp. 820–826, Nov. 1977.
22. ROBINSON, G.: Color Edge Detection, *Optical Engineering*, Vol. 16, No. 5, pp. 479–484, Sept./Oct. 1987.
23. OHTA, Y.: Knowledge-based Interpretation of Outdoor Natural Color Scenes, Pitman, Advanced Publishing Programm, Boston, 1985.
24. OHTA, Y. – KANADE, T. – SAKAI, T.: Color Information for Region Segmentation, *Computer Vision Graphics and Image Processing*, Vol. 13, pp. 222–241, 1980.
25. SARABI, A. – AGGARWAL, J. K.: Segmentation of Chromatic Images, *Pattern Recognition*, Vol. 13, No. 6, pp. 417–427, 1981.
26. CHASSERY, J. M. – GARBAY, C.: An Iterative Segmentation Method Based on Contextual Color and Shape Criterion, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-6, No. 6, pp. 794–800, Nov. 1984.
27. TOMINAGA, S.: Color image Segmentation Using Three Perceptual Attributes, *Proceedings of the IEEE International Conference on Computer Vision*, pp. 628–630, 1986.
28. THERRIEN, C.: Multichannel Filtering Methods for Color Image Segmentation, *Proceedings of the IEEE International Conference on Pattern Recognition*, pp. 637–639, 1985.
29. LIMB, J. – RUBINSTEIN C. B. – THOMPSON, J. E.: Digital Coding of Video Signals: a Review, *IEEE Trans. on Communications*, Vol. COM-25, pp. 1349–1384, Nov. 1977.
30. RZESZEWSKI, T. (Ed.): Television Technology Today, IEEE Press, 1985.
31. FUJIO, T.: High Definition Television Systems: Desirable Standards, Signal Forms and Transmission Systems, *IEEE Trans. on Communications*, Vol. COM-29, pp. 1882–1891, Dec. 1981.
32. ARONOFF, S. – JONES, G. F.: From Data to Image Action, *IEEE Spectrum*, pp. 45–52, Dec. 1985.
33. VENETSANOPOULOS, A. N.: Digital Image Processing and Analysis, in Signal Processing, (Durrani and Lacoume, Eds.), Amsterdam, North Holland, 1986.
34. BALLARD, D. H. – BROWN, C. M.: Computer Vision, Englewood Cliffs, NJ, Prentice Hall, 1982.
35. KUNG, S. Y.: VLSI Array Processors, Englewood Cliffs, NJ, Prentice Hall, 1987.
36. HWANG, K.: Advanced Computer Architecture: Parallelism, Scalability, Programmability, New York, McGraw-Hill, 1993.
37. KUNG, H. T.: Why Systolic Architectures, *IEEE Computer*, Vol. 15, pp. 37–46, 1982.
38. KUNG, S. Y.: On Supercomputing with Systolic/Wavefront Array Processors, *Proc. IEEE*, Vol. 72, pp. 867–884, 1984.
39. LU, H. H. – LEE, E. A. – MESSERSCHMITT, D. G.: Fast Recursive Filtering with Multiple Slow Processing Elements, *IEEE Trans. on Circuits and Systems*, Vol. CAS-32, pp. 1119–1129, Nov. 1985.
40. PARHI, K. K. – MESSERSCHMITT, D. G.: Concurrent Cellular VLSI Adaptive Filter Architectures, *IEEE Trans. on Circuits and Systems*, Vol. CAS-34, pp. 1141–1151, 1987.
41. MERTZIOS, B. G. – SYRMOS, V. L.: Implementation of Digital Filtering via VLSI Array Processors, *IEE Proc., Part G*, Vol. 135, pp. 78–82, 1988.

42. MERTZIOS, B. G. – SYRMOS, V. L.: VLSI Array Processors Block Implementation of IIR Digital Filters, *Circuits, Systems and Signal Processing*, Vol. 7, No. 1, pp. 79–94, 1988.
43. MERTZIOS, B. G. – VENETSANOPOULOS, A. N.: Implementation of Quadratic Digital Filters via VLSI Array Processors, *Archiv für Elektronik und Übertragungstechnik, AEU*, Vol. 43, No. 3, pp. 153–157, 1989.
44. MERTZIOS, B. G.: Fast Block Implementation of Two-dimensional Recursive Digital Filters via VLSI Array Processors, *Archiv für Elektronik und Übertragungstechnik, AEU*, Vol. 44, No. 1, pp. 50–58, 1990.
45. BURRUS, C. S.: Block Realization of Digital Filters, *IEEE Trans. on Electroacoust.*, Vol. AU–20, pp. 230–235, 1972.
46. MERTZIOS, B. G.: Block Realization of 2–D IIR Digital Filters, *Signal Processing*, Vol. 7, pp. 135–149, 1984.
47. MERTZIOS, B. G.: Block Parallel Processing of 2–D Digital Signals, *Circuit Theory and Applications*, Vol. 14, pp. 211–228, 1986.
48. LEE, J. H. – WOODS, J. W.: Sectioned Implementation of Two-dimensional Half-plane Recursive Filters, *IEEE Trans. on Acoust., Speech and Signal Processing*, Vol. ASSP–33, pp. 1272–1279, 1985.
49. VENETSANOPOULOS, A. N. – MERTZIOS, B. G.: A Decomposition Theorem and its Applications to the Design and Realization of Two-dimensional Filters, *IEEE Trans. on Acoust., Speech and Signal Processing*, Vol ASSP–33, pp. 1562–1574, December 1985.
50. MERTZIOS, B. G. – VENETSANOPOULOS, A. N.: Modular Realization of m-dimensional Filters, *Signal Processing*, Vol. 7, pp. 351–369, 1984.
51. MERTZIOS, B. G. – VENETSANOPOULOS, A. N.: Block Decomposition Structures for the Fast Modular Implementation of Two-dimensional Digital Filters, *Circuits, Systems and Signal Processing*, Vol. 8, No. 2, pp. 163–186, 1989.
52. MERTZIOS, B. G. – VENETSANOPOULOS, A. N.: Walsh-Hadamard Block Decomposition of FIR Two-dimensional Digital Filters, *Int. J. of Electronics*, Vol. 68, No. 6, pp. 991–1004, 1990.
53. ELLIOT, D. F. – RAO, R.: Fast Transforms, Algorithms Analyses and Applications, London: Academic Press, 1982.
54. SCHETZEN, M.: The Volterra and Wiener Theories of Nonlinear Systems, New York, J. Wiley, 1980.
55. MERTZIOS, B. G. – SICURANZA, G. L. – VENETSANOPOULOS, A. N.: Efficient Realizations of Two-dimensional Quadratic Digital Filters, *IEEE Trans. on Acoust., Speech and Signal Processing*, Vol. ASSP–37, pp. 765–769, May 1989.
56. MERTZIOS, B. G. – VENETSANOPOULOS, A. N.: Fast Implementation of Nonlinear Volterra Digital Filters via Systolic Arrays, *Proceedings of the 1990 Bilkent International Conference on New Trends in Communications, Control and Signal Processing*, pp. 1128–1134, July 2–5, 1990, Ankara, Turkey.