A Fast Filter for Real-Time Image Processing

KICH M. TY, STUDENT MEMBER, IEEE, AND
ANASTASIOS N. VENETSANOPOULOS, SENIOR MEMBER, IEEE

Abstract—**In this paper, a new digital filter structure is developed for the implementation of two-dimensional (2-D) recursive filters for real-time image processing. The proposed structure has a short clock cycle time or a high data throughput rate, independent of the order of the filter. Parallelism and pipelining are the two features of the proposed filter structure that contribute to its high-speed performance. The filter can be implemented without multipliers. Using standard integrated circuits and memories, the new filter is capable of filtering images of size up to 512×512 pixels with a TV scan rate of 30 frames/s in real time. The effects of the finite precision arithmetic have been considered. Scaling and overflow problems are studied to give insight into the choice of a proper scaling factor, so that an adequate signal-to-noise ratio at the filter output can be obtained.**

I. Introduction

IN THE PAST FEW years, real-time image processing using two-dimensional (2-D) digital filters has become a rapidly growing field in the industrial and biomedical environments, as the need for the fast processing of large amounts of data became evident. The term "real-time image processing" can be defined as "the processing of images at a speed such that the data rate of the processed images is the same as that of the input images." If one considers an image of size $M \times N$ pixels and a TV scan rate of L frames/s, and if R arithmetic operations are required for each output pixel, the total number of arithmetic operations that have to be performed in one second is $M \times N \times L \times R$ [1]. We now summarize a number of filter structures that are already known to operate at high speed.

Peled and Liu [2] described an implementation of digital filters by distributed arithmetic. This method has proved its advantage with respect to speed, cost, and power dissipation by storing all possible binary sums of the filter coefficients in a programmable read only memory (PROM). Distributed arithmetic implementation of 2-D digital filters can be found in [3] and [4]. Where the variability of the filter coefficients is important, the Canonical Sign Digit (CSD) representation of the filter coefficients [5] and the use of stored square ROM [6] have been suggested. Residue Number System Arithmetic results in a highly parallel hardware design with characteristically high computational speed [7]. More recently, a new memory-oriented implementation of 2-D digital filters with each coefficient expressed by an algebraic sum of power-of-a terms has been presented [8]. The stored product digital filter architecture as formulated in [9] and [10] presents another alternative for the elimination of the multipliers through the use of ROM's.

A general configuration for 1-D recursive digital filters [11] has shown that high-speed, 10 MHz or higher, word throughput rates for parallel operations in two's complement fixed point arithmetic are feasible with reasonable memory size and standard logic devices. The high operating speed of the filter is due to its parallel–pipelined structure. This approach is different from the other implementation schemes, in the sense that "a minimum number of arithmetic operations are required in one clock cycle." If we consider an input data rate of S samples/s, the quantity $1/S$ is the duration of one clock cycle. The main advantage of this method is that the data throughput rate is independent of the order of the filter.

The purpose of this paper is to present a 2-D recursive digital filter structure (also valid for nonrecursive digital filters) that can operate at a very high speed. Section II describes the detailed development of the new 2-D digital filter structure that has a short critical path, from both the theoretical and practical points of view. Section III contains the hardware description of a second-order 2-D recursive filter for filtering of images of size up to 512×512 pixels with a TV scan rate of 30 frames/s in real time. In Section IV, error analysis of the new 2-D filter structure is presented. Scaling, overflow problem, and experimental results with real images are covered in Section V. Section VI, finally, presents a summary of contributions made in this research.

II. High Speed Two-Dimensional Recursive Digital Filter

A 2-D causal recursive digital filter is described by the linear difference equation

$$y_{m,n} = \sum_{i=0}^{M_A} \sum_{j=0}^{N_A} a_{i,j} x_{m-i,n-j} - \sum_{\substack{k=0 \\ k+l}}^{M_B} \sum_{\substack{l=0 \\ \neq 0}}^{N_B} b_{k,l} y_{m-k,n-l} \quad (1)$$

where x and y are the input image arrays, respectively, and $a_{i,j}$ and $b_{k,l}$ are the filter coefficients of the nonrecursive and recursive blocks, respectively. M_A and N_A de-

Manuscript received July 31, 1985, revised February 28, 1986. This work was supported in part by an NSERC grant.
The authors are with the Department of Electrical Engineering, University of Toronto, Toronto, Ontario, Canada M5S 1A4.
IEEE Log Number 8609851.

0098-4094/86/1000-0948$01.00

scribe the size of the input mask, whereas M_B and N_B describe the size of the output mask.

Some of the 2-D recursive digital filters described by (1) which claim the capability of filtering images in real-time have a structure similar to that as shown in Fig. 1, this kind of filter structure will be referred to as "direct form" implementation. The structure is so named because the output of the filter, $y_{m,n}$, is computed directly from the difference equation (1). Although this filter structure exhibits high parallelism, its main disadvantage is the limitation in speed due to the propagation of the intermediate results through the adder tree in the computational unit. Our aim is to replace the adder tree by a more rational design of 2-D recursive digital filter structure so that the new filter will have a higher operating speed [12]–[14].

The causal digital filter described by (1) has the transfer function

$$H(Z_1, Z_2) = \frac{\sum_{i=0}^{M_A} \sum_{j=0}^{N_A} a_{i,j} Z_1^{-i} Z_2^{-j}}{1 + \sum_{\substack{k=0 \\ k+l \neq 0}}^{M_B} \sum_{l=0}^{N_B} b_{k,l} Z_1^{-k} Z_2^{-l}} \tag{2}$$

where Z_1^{-1} and Z_2^{-1} represent row and column delays, respectively. Denote the numerator of the filter transfer function by $N(Z_1, Z_2)$ and the denominator by $D(Z_1, Z_2)$. Using Horner's rule [15], $N(Z_1, Z_2)$ can equivalently be expressed as

$$\begin{aligned} N(Z_1, Z_2) = a_{0,0} &+ Z_2^{-1}\left(a_{0,1} + \cdots + Z_2^{-1}(a_{0,N_A})\cdots\right) \\ &+ Z_1^{-1}\Big[a_{1,0} + Z_2^{-1}\left(a_{1,1} + \cdots\right. \\ &+ Z_2^{-1}(a_{1,N_A})\cdots\Big) + \cdots \\ &+ Z_1^{-1}\Big[a_{M_A,0} + Z_2^{-1}\left(a_{M_A,1} + \cdots\right. \\ &+ Z_2^{-1}(a_{M_A,N_A})\cdots\Big)\Big]\cdots\Big]. \end{aligned} \tag{3}$$

Similarly, $D(Z_1, Z_2)$ can also expressed as

$$\begin{aligned} D(Z_1, Z_2) = 1 &+ Z_2^{-1}\left(b_{0,1} + \cdots + Z_2^{-1}(b_{0,N_B})\cdots\right) \\ &+ Z_1^{-1}\Big[b_{1,0} + Z_2^{-1}\left(b_{1,1} + \cdots\right. \\ &+ Z_2^{-1}(b_{1,N_B})\cdots\Big) + \cdots \\ &+ Z_1^{-1}\Big[b_{M_B,0} + Z_2^{-1}\left(b_{M_B,1} + \cdots\right. \\ &+ Z_2^{-1}(b_{M_B,N_B})\cdots\Big)\Big]\cdots\Big]. \end{aligned} \tag{4}$$

The schematic block diagrams of $N(Z_1, Z_2)$ and $1/D(Z_1, Z_2)$ for a second-order 2-D digital filter are shown in Fig. 2a and 2b, respectively. Both $N(Z_1, Z_2)$ and $1/D(Z_1, Z_2)$ have a transposed structure [16]. The cascade of these two blocks will result in the original filter transfer function given by (2).

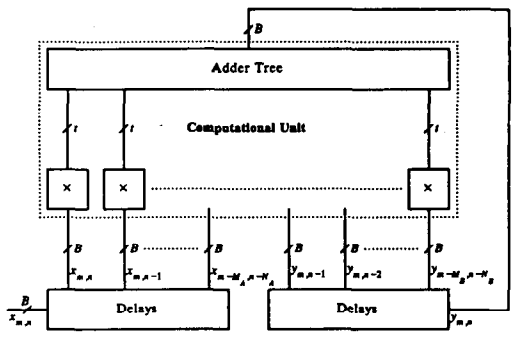
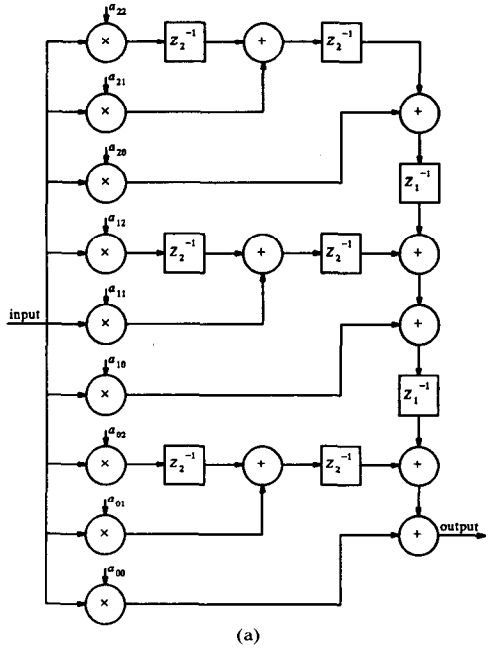
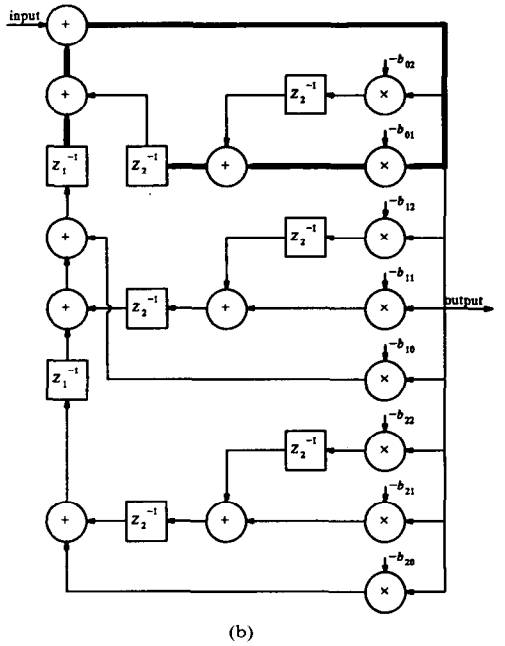


Fig. 1. *Direct form* implementation of a 2-D recursive digital filter. × —multiplier, B—number of bits used for the input/output, t—number of bits used for the intermediate results.



(a)



(b)

Fig. 2. (a) Nonrecursive block $N(Z_1, Z_2)$ of the second-order 2-D digital filter. (b) Recursive block $1/D(Z_1, Z_2)$ of a second-order 2-D digital filter.

All signals are represented in fixed point two's complement code. The input samples and the output signals, assumed to be bounded by ± 1, have B bits of accuracy including the sign bit. All other intermediate results have t bits of accuracy with $t > B$. Thus, the first block of the cascaded structure has B input bits and t output bits, while the second block has t input bits and B output bits. There are two ways of cascading these two blocks. The criteria is to design a filter structure with a short critical path, which is defined as the longest path among all the possible paths from the output of delay element to the input of the next one. Thus, the maximum operating speed of any filter is determined by the length of its critical path. The critical path contains one multiplication and three additions when the recursive block is followed by the nonrecursive block. This path is shown in Fig. 2b in bold lines. When the two blocks are interchanged, the critical path contains two multiplications and three additions. It is possible to further reduce the number of arithmetic operations in the critical path, as in the 1-D case [11]. The minimum number of arithmetic operations required in the critical path of a 2-D recursive digital filter has to be figured out first.

For 1-D digital filters, a configuration for which the critical path contains no more than one multiplication and one addition has been derived [11]. For 2-D digital filters, the critical path should contain only one more addition than that of the 1-D filter. The extra addition is required for adding the intermediate results from the Z_1^{-1} and Z_2^{-1} blocks. In order to obtain the desired critical path, the original 2-D filter transfer function should be modified so that the new transfer function $\hat{H}(Z_1, Z_2)$ should have the form

$$\hat{H}(Z_1, Z_2) = H(Z_1, Z_2) Z_1^{-p} Z_2^{-q} \tag{5}$$

with p and q being nonnegative integers. The use of latency makes it possible to complete all the arithmetic operations required for an output in more than one clock cycle, thus resulting in a shorter clock cycle. It is desirable to have the minimum possible latency, which is defined as the time interval separating the appearance of an input sample at the input port from the appearance of the corresponding output at the output port. This can be achieved by setting $p = 1$ and $q = 0$. Thus, the output of new filter $\hat{y}_{m,n}$ is delayed by only one pixel compared with the original filter output $y_{m,n}$ (i.e., $\hat{y}_{m,n} = y_{m-1,n}$). With the chosen values for p and q, the new equation describing the input and output relationship in the 2-D z-domain is

$$\hat{Y}(Z_1, Z_2) = Z_2^{-1} X(Z_1, Z_2) \sum_{i=0}^{M_A} \sum_{j=0}^{N_A} a_{i,j} Z_1^{-i} Z_2^{-j} - \hat{Y}(Z_1, Z_2) \sum_{\substack{k=0 \\ k+l \neq 0}}^{M_B} \sum_{l=0}^{N_B} b_{k,l} Z_1^{-k} Z_2^{-l}. \tag{6}$$

We now propose the new filter structure for the modified 2-D filter transfer function. Let

$$A = X(Z_1, Z_2) \sum_{\substack{i=0 \\ i+j \neq 0}}^{M_A} \sum_{j=0}^{N_A} a_{i,j} Z_1^{-i} Z_2^{-j} \tag{7}$$

$$B = -\hat{Y}(Z_1, Z_2) \sum_{i=2}^{M_B} b_{0,l} Z_2^{-(l-1)} \tag{8}$$

$$C = a_{0,0} X(Z_1, Z_2) \tag{9}$$

$$D = A + B + C \tag{10}$$

$$E = -b_{0,1} \hat{Y}(Z_1, Z_2) \tag{11}$$

$$F = Z_2^{-1}(D + E) \tag{12}$$

$$G = -\hat{Y}(Z_1, Z_2) \sum_{k=1}^{M_B} \sum_{l=0}^{N_B} b_{k,l} Z_1^{-k} Z_2^{-l} \tag{13}$$

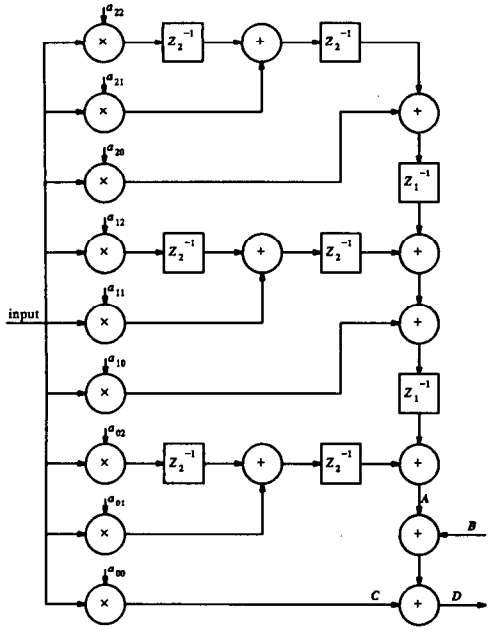
$$I = F + G. \tag{14}$$

The relationships among all these signals of a second-order filter are shown in Fig. 3. It can easily be proven that $\hat{Y}(Z_1, Z_2) = I$. With the assumption that a multiplication takes at least twice the amount of time required for an addition, the critical path is the one shown in bold lines and contains only one multiplication and two additions. Moreover, this critical path is independent of the order of the filter. The new filter has a very regular structure, with identical building blocks. This regularity property provides a simple hardware structure for the implementation of the filter. The new filter also has a small hardware size since the input to the multiplier block has only B bits.

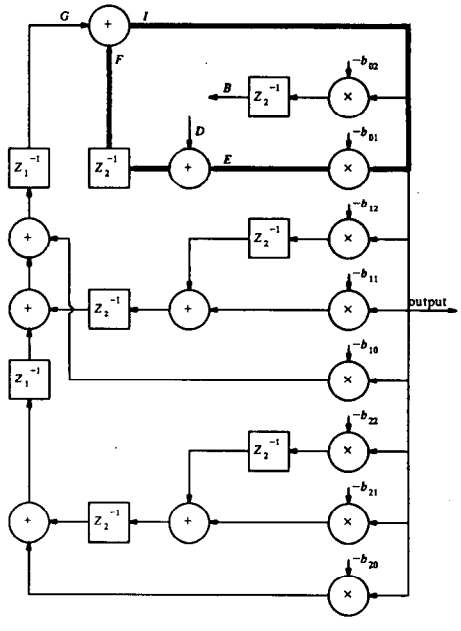
III. Hardware for Real-Time Image Processing

Consider the processing of an image of size 512×512 pixels with a TV scan rate of 30 frames/s in real time, the required data throughput rate S is $512 \times 512 \times 30 = 7.86 \times 10^6$ pixels/s or one pixel every 127 ns. For a reasonable gray level resolution, the input and the final output signals are represented in $B = 8$ bits. All intermediate results have $t = 16$ bits of accuracy. The hardware of the new filter of second order with the above specifications will be outlined in this section.

The new parallel–pipelined structure of a 2-D recursive digital filter consists mainly of three building blocks: 1) delay units, 2) multipliers, and 3) adders/subtractors. Input to the multipliers of the nonrecursive block is the serial sampled video data resulting from the raster scan of an image of size 512×512 pixels, whereas the input to the multipliers of the recursive block is the most recently computed output. High-speed multipliers are very expensive and are not economical for the implementation of fast filters. One method of replacing the multipliers, namely the "stored product" method [11], is considered in the hardware implementation.



(a)



(b)

Fig. 3. (a) Modified nonrecursive block of a second-order 2-D digital filter. (b) Modified recursive block of a second-order 2-D digital filter.

The Z_1^{-1} delay elements can be configured from SN 74S174 (hex D-type flip-flops), which has a maximum propagation delay of 17 ns and a minimum set up time of 5 ns [17]. The Z_2^{-1} delay elements, which consist of 16 parallel 512-bit shift registers, can be configured from TDC 1006J (1×256) having a maximum propagation delay of 30 ns and a minimum setup time of 0 ns. It requires two TDC 1006J chips for the implemetation of one 512-bit shift register. The number of IC chips required for SN 74S174 and TDC 1006J are 32 and 128, respectively. Since the sum of the propagation delay and the set up time of SN 74S174 is shorter than that of TDC 1006J, it is desirable to drive two different shift registers with two different clock signals, one being the delayed version of the other, so that the outputs of the two different shift registers will be available at almost the same instant. From the specifications of the shifts registers, the clock signal driving the single-bit shift register should be the one driving the 512-bit shift register delayed by 5 to 13 ns.

Each 16-bit adder is constructed from four 74LS181 4-bit ALU's and one 74LS182 carry look ahead generator. Addition of two 16-bit binary numbers takes 19 ns [18]. The 16 adders, required for the 16 additions in a second-order 2-D difference equation, take $16\times5=80$ IC chips.

Using the *stored product* method, the multipliers are replaced by memories. The memories, if constructed from AM 27S20 (256×4) PROM's, which have an access time of 45 ns [18], would require 36 packages and 32 packages for the nonrecursive block and the recursive block, respectively.

The cycle time of the new filter, built with the above components, consists of one memory access time, two addition times, and the sum of the propagation delay and setup time of the 512-bit shift register. The new filter can process images at a data throughput rate of one pixel every 113 ns (i.e., $45+19\times2+30$ ns), which is less than the maximum allowable time (127 ns) required for real-time processing.

In addition to the cycle time, another measurement for the filter performance is the latency. The latency for the proposed filter structure is the sum of the following (see Fig. 3):

i) one cycle time (127 ns for the processing of an image of size 512×512 pixels with a TV scan rate of 30 frames/s).
ii) propagation delay of the 512-bit shift registers (30 ns), and
iii) one addition time (19 ns).

This latency of 176 ns is independent of the order of the filter, which is another attractive feature of the new filter. A summary of the hardware and throughput rate of the new filter structure is shown in Tables I and II.

IV. Error Analysis

The effects of finite precision are considered in the new 2-D recursive digital filter. Errors are introduced in quantizing the input and in the roundoff accumulation of the intermediate results. In this section, and analysis of the steady-state statistics of such errors is presented. The analysis is based on recursive implementation, but the results for nonrecursive implementation can also be obtained by specializing the obtained results. Two's complement fixed point arithmetic is used and the distinction between rounding and truncation is made.

TABLE I
HARDWARE COMPONENTS AND COSTS OF STORED PRODUCT IMPLEMENTATION SCHEMES OF A SECOND-ORDER 2-D RECURSIVE DIGITAL FILTER FOR THE PROCESSING OF IMAGES OF SIZES 512×512 PIXELS

Components	Part Number	Organization	# of Ic's	Estimated Cost 1984 (U.S. $)
PROM's 256×16	AM27S20	256×4	68	$170
ALU's	SN74S181	—	64	$256
Carry Look Ahead Generator	SN74S182	—	16	$32
Shift Register 1×512	TDC1006J	1×256	128	$5120
1×1	SN74S174	Hex D Flip-flops	32	$50
		Total	308	$5628

TABLE II
CYCLE TIME AND LATENCY OF THE STORED PRODUCT IMPLEMENTION

T_c	Cycle time
T_l	Latency
T_a	Time for 16-bit addition
T_{ma}	Memory access time
T_s	512-bit shift register setup time
T_{pd}	512-bit shift register propagation delay time

T_c	$T_{ma}+2T_a+T_s+T_{pd}$ $=45+2\times19+0+30$ ns $=113$ ns <127 ns
T_c increases with the Order of the Filter	No
T_l	$127\text{ ns}+T_s+T_{pd}+T_a$ $=127+0+30+19$ ns $=176$ ns
T_l increases with the Order of the Filter	No

To simplify the analysis, some assumptions about the statistical properties of the quantization errors are made.

i) The sequence of error samples is a sample sequence of a stationary random process.

ii) The quantization process is white. The random variables representing the error process are uncorrelated, independent of the sampling rate.

iii) The error sequence is uncorrelated with the sequence of exact samples.

iv) The quantization error has a unifrom density function. This implies that the signal is equally likely to be anywhere within a quantization interval.

v) Overflow does not occur at the output of the filter.

Quantization errors are caused by either truncation or rounding, each mode resulting in a different error effect. The filter output error is independent of the factor Z_1^{-1} associated with the modified filter transfer and, for simplicity, we will use the original difference equation (1) describing the 2-D recursive digital filter in the analysis.

With the *stored product* method, the finite precision representation of (1) becomes

$$\bar{y}_{m,n}=\left[\sum_{i=0}^{M_A}\sum_{j=0}^{N_A}(a_{i,j}\bar{x}_{m-i,n-j})_t+\sum_{\substack{k=0\\k+l}}^{M_B}\sum_{\substack{l=0\\\neq 0}}^{N_B}(-b_{k,l}\bar{y}_{m-k,n-l})_t\right]_B \tag{15}$$

where $(*)_t$ and $[*]_B$ represent the quantization of $*$ to t bits and B bits of precision, respectively. The coefficient quantization error is not present because all coefficients can still in very high precision, instead of t bits, before multiplications. All products and sums are represented in t bits and final output is quantized to B bits. Denoting the errors introduced by $(a_{i,j}\bar{x}_{m-i,n-j})_t$, $(-b_{k,l}\bar{y}_{m-k,n-l})_t$, and $[*]_B$ by $\epsilon^t_{i,j}$, $\epsilon^t_{k,l}$ and $\epsilon^{t,B}$, respectively, (15) becomes

$$\bar{y}_{m,n}=\sum_{i=0}^{M_A}\sum_{j=0}^{N_A}\left(a_{i,j}\bar{x}_{m-i,n-j}+\epsilon^t_{i,j}\right)+\sum_{\substack{k=0\\k+l}}^{M_B}\sum_{\substack{l=0\\\neq 0}}^{N_B}\left(-b_{k,l}\bar{y}_{m-k,n-l}+\epsilon^t_{k,l}\right)+\epsilon^{t,B}. \tag{16}$$

Let $e_{m,n}$ be the quantization error in the input and $f_{m,n}$ be the error in the output with

$$e_{m,n}=\bar{x}_{m,n}-x_{m,n} \tag{17a}$$

$$f_{m,n}=y_{m,n}-\bar{y}_{m,n}. \tag{17b}$$

Finally, (1), (16), (17a), and (17b) result in

$$f_{m,n}=-\sum_{i=0}^{M_A}\sum_{j=0}^{N_A}a_{i,j}e_{m-i,n-j}-\sum_{i=0}^{M_A}\sum_{j=0}^{N_A}\epsilon^t_{i,j}-\sum_{\substack{k=0\\k+l}}^{M_B}\sum_{\substack{l=0\\\neq 0}}^{N_B}\epsilon^t_{k,l}-\epsilon^{t,B}-\sum_{\substack{k=0\\k+l}}^{M_B}\sum_{\substack{l=0\\\neq 0}}^{N_B}b_{k,l}f_{m-k,n-l}. \tag{18}$$

The first term describes the effect of input quantization error. The second, third, and fourth terms describe the roundoff accumulation error. The uncorrelated assumption

allows the total error to be considered as a linear combination of two independent sources of errors.

The statistics of the total error $f_{m,n}$ are found to be [19]

$$E[f] = -E[e]\frac{\sum_{i=0}^{M_A}\sum_{j=0}^{N_A} a_{i,j}}{\sum_{k=0}^{M_B}\sum_{l=0}^{N_B} b_{k,l}} + \frac{1}{\sum_{k=0}^{M_B}\sum_{l=0}^{N_B} b_{k,l}} \cdot \left[-\sum_{i=0}^{M_A}\sum_{j=0}^{N_A} E\left[\epsilon_{i,j}^{t}\right] - \sum_{\substack{k=0 \\ k+l}}^{M_B}\sum_{\substack{l=0 \\ \neq 0}}^{N_B} E\left[\epsilon_{k,l}^{t}\right] - E\left[\epsilon^{t,B}\right] \right] \tag{19a}$$

$$\sigma_f^2 = \sigma_e^2 \sum_{m=0}^{\infty}\sum_{n=0}^{\infty} h_{m,n}^2 + \left[\sum_{m=0}^{\infty}\sum_{n=0}^{\infty} d_{m,n}^2\right] \cdot \left[\sum_{i=0}^{M_A}\sum_{j=0}^{N_A} \sigma_{\epsilon_{i,j}^{t}}^2 + \sum_{\substack{k=0 \\ k+l}}^{M_B}\sum_{\substack{l=0 \\ \neq 0}}^{N_B} \sigma_{\epsilon_{k,l}^{t}}^2 + \sigma_{\epsilon^{t,B}}^2 \right] \tag{19b}$$

where $E[f]$ and $\sigma_f{}^2$ are the mean and the variance of the total error at the output of the filter, and $d_{m,n}$ is the inverse z-transform of $1/D(Z_1, Z_2)$.

If all the coefficients of the nonrecursive block $a_{i,j}$'s are multiplied by a scaling factor ρ, the new statistics of the total error at the output of the filter are given by

$$E[f] = -\rho E[e]\frac{\sum_{i=0}^{M_A}\sum_{j=0}^{N_A} a_{i,j}}{\sum_{k=0}^{M_B}\sum_{l=0}^{N_B} b_{k,l}} + \frac{1}{\sum_{k=0}^{M_B}\sum_{l=0}^{N_B} b_{k,l}} \cdot \left[-\sum_{i=0}^{M_A}\sum_{j=0}^{N_A} E\left[\epsilon_{i,j}^{t}\right] - \sum_{\substack{k=0 \\ k+l}}^{M_B}\sum_{\substack{l=0 \\ \neq 0}}^{N_B} E\left[\epsilon_{k,l}^{t}\right] - E\left[\epsilon^{t,B}\right] \right] \tag{20a}$$

$$\sigma_f^2 = \rho^2\sigma_e^2 \sum_{m=0}^{\infty}\sum_{n=0}^{\infty} h_{m,n}^2 + \left[\sum_{m=0}^{\infty}\sum_{n=0}^{\infty} d_{m,n}^2\right] \cdot \left[\sum_{i=0}^{M_A}\sum_{j=0}^{N_A} \sigma_{\epsilon_{i,j}^{t}}^2 + \sum_{\substack{k=0 \\ k+l}}^{M_B}\sum_{\substack{l=0 \\ \neq 0}}^{N_B} \sigma_{\epsilon_{k,l}^{t}}^2 + \sigma_{\epsilon^{t,B}}^2 \right] \tag{20b}$$

respectively. Thus, a fixed amount of distortion is always present in the output of the filter no matter what the scaling factor ρ is.

Theoretical and simulation results of the error at the output of the *stored product* 2-D recursive digital filters were obtained. Double precision arithmetic was used for the simulation of the ideal (infinite precision) filter. The main advantage of using computer simulation to compute the statistics of the quantization error is the possibility of studying the rounding effects, the truncation effects, and the scaling effects separately, with few changes in the filtering algorithm.

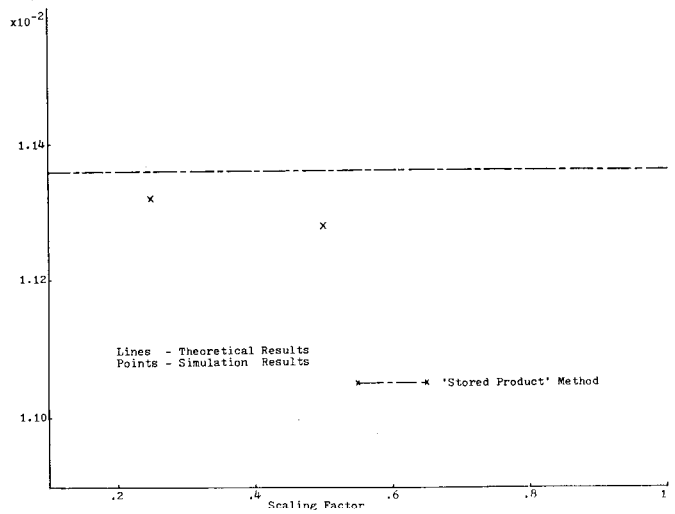


Fig. 4. Mean of the error at the output of filter #1 using the *stored product* method (rounding).

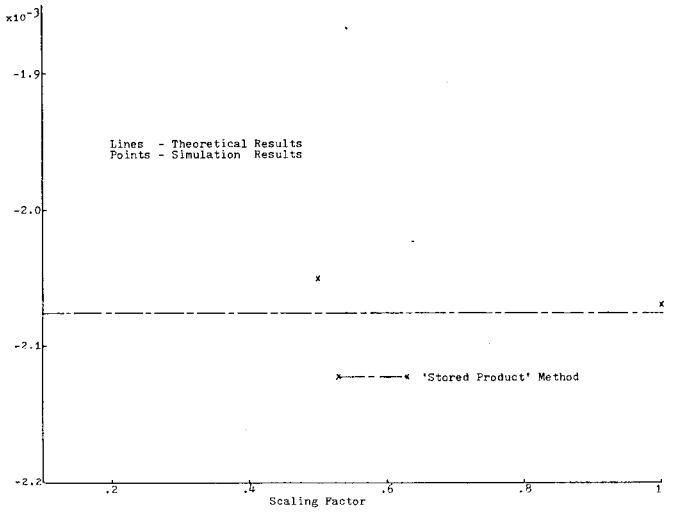


Fig. 5. Mean of the error at the output of filter #2 using the *stored product* method (rounding).

Deterministic signals and random signals were used as input images. A first-order (filter #1) and a second-order (filter #2) 2-D filter, drawn from [20], of the form

$$H(Z_1, Z_2) = \rho\frac{\sum_{i=0}^{M_A}\sum_{j=0}^{N_A} a_{i,j}Z_1^{-i}Z_2^{-j}}{\sum_{k=0}^{M_B}\sum_{l=0}^{N_B} b_{k,l}Z_1^{-k}Z_2^{-l}} \tag{21}$$

were used for simulation, with ρ being the scaling factor. The specifications of the two filters are given in Tables III and IV. The inputs and final outputs can each be truncated or rounded. Two sets of simulations were carried out; the first one with all signals and results truncated, whereas the second one with feedback truncated and all other signals rounded. The results of the second set of simulations are shown in Figs. 4–6. It was verified that the theoretical results agreed very well with the simulated results for $t = 16$, $B = 8$. The main sources of the filter

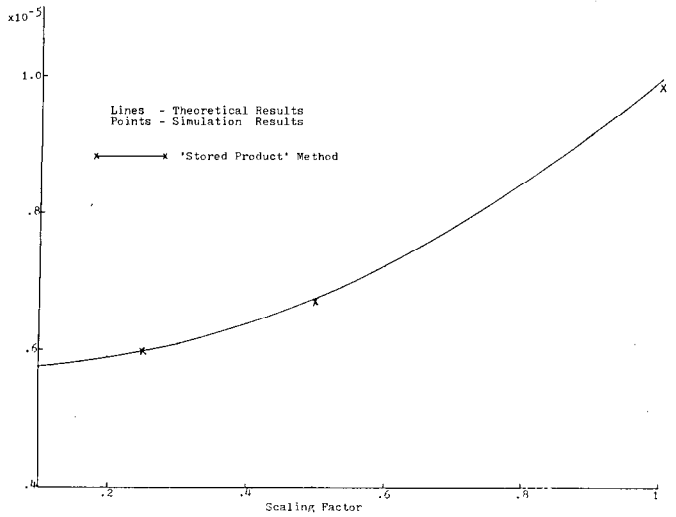


Fig. 6. Variance of the error at the output of filter #2 using the *stored product* method.

TABLE III
SPECIFICATIONS OF FILTER #1

Coefficients of the Numerator	Coefficients of the Denominator
$a_{0,0}=1$	$b_{0,0}=1$
$a_{0,1}=-0.04668$	$b_{0,1}=-0.42602$
$a_{1,0}=-0.04668$	$b_{1,0}=-0.42602$
$a_{1,1}=-0.46761$	$b_{1,1}=0.10692$
$\sum_{m=0}^{\infty}\sum_{n=0}^{\infty} h_{m,n}^2$	1.43075
$\sum_{m=0}^{\infty}\sum_{n=0}^{\infty} \lvert h_{m,n}\rvert$	2.93811
$\sum_{m=0}^{\infty}\sum_{n=0}^{\infty} d_{m,n}^2$	1.58463

output error are the input quantization error and roundoff accumulation error, especially the quantization of the 16-bit results to 8 bits in the final output and the feedback.

V. SCALING OF THE TRANSFER FUNCTION

If the amplitude of the output signal of a recursive digital filter in a fixed point implementation is allowed to exceed the dynamic range, overflow will occur and the output signal will be severely distorted. This is due to the fact that output error due to overflow is fedback into the recursive filter. On the other hand, if the output signal amplitude is unduly low, the filter is operating inefficiently, and the signal-to-noise ratio will be poor. Therefore, for optimum filter performance, suitable scaling must be employed to adjust the output signal levels.

A 2-D recursive difference equation can always be written as

$$y_{m,n}=\rho\sum_{k=0}^{\infty}\sum_{l=0}^{\infty} h_{k,l}x_{m-k,n-l} \tag{22}$$

where ρ is a positive scaling factor. Evidently, the magnitude of the output signal $|y_{m,n}|$ is

$$\begin{aligned}|y_{m,n}| &= \rho\left|\sum_{k=0}^{\infty}\sum_{l=0}^{\infty} h_{k,l}x_{m-k,n-l}\right| \\ &\leqslant \rho\sum_{k=0}^{\infty}\sum_{l=0}^{\infty}|h_{k,l}x_{m-k,n-l}| \\ &= \rho\sum_{k=0}^{\infty}\sum_{l=0}^{\infty}|h_{k,l}||x_{m-k,n-l}|. \end{aligned}\tag{23}$$

If $|x_{m,n}|\leqslant 1$ for all m and n, the magnitude of the output signal is given by

$$|y_{m,n}|\leqslant\rho\sum_{k=0}^{\infty}\sum_{l=0}^{\infty}|h_{k,l}|. \tag{24}$$

To ensure absolutely no overflow in the output, i.e., $|y_{m,n}|\leqslant 1$, scaling factor ρ must satisfy the condition

$$\rho\sum_{k=0}^{\infty}\sum_{l=0}^{\infty}|h_{k,l}|\leqslant 1. \tag{25}$$

or

$$\rho\leqslant\frac{1}{\sum_{k=l}^{\infty}\sum_{l=0}^{\infty}|h_{k,l}|}. \tag{26}$$

We don't have to worry about the overflows that can occur in the intermediate results. This is due to the fact that if $y_{m,n}$ has no overflow, it is always evaluated correctly in two's complement arithmetic, even if overflows do occur in the partial sums.

In this section, we are going to present some experimental results of the filtering real images using two different filters scaled by different scaling factors. The main objective of these filtering experiments is to estimate an optimum scaling factor for given 2-D recursive digital filter used for image processing.

The filtering process is simulated on VAX 11/780 computer. Only the coefficients of the nonrecursive block are scaled by ρ. The scaling of an input might cause an overflow at the input port of the filter. The first filter used

TABLE IV
SPECIFICATIONS OF FILTER #2

Coefficients of the Numerator	Coefficients of the Denominator
$a_{0,0}=-0.1557553\times10^{-1}$	$b_{0,0}=1$
$a_{0,1}=0.468344\times10^{-1}$	$b_{0,1}=0.223576$
$a_{0,2}=-0.399411\times10^{-2}$	$b_{0,2}=0.7149619\times10^{-1}$
$a_{1,0}=0.4951426\times10^{-1}$	$b_{1,0}=0.22108698$
$a_{1,1}=-0.2103131\times10^{-1}$	$b_{1,1}=0.1544512$
$a_{1,2}=-0.2237115$	$b_{1,2}=0.1057191$
$a_{2,0}=0.411281\times10^{-2}$	$b_{2,0}=0.9173054\times10^{-1}$
$a_{2,1}=-0.2336235$	$b_{2,1}=0.1020029$
$a_{2,2}=0.707513$	$b_{2,2}=0.15625$
$\sum_{m=0}^{\infty}\sum_{n=0}^{\infty} h_{m,n}^2$	0.823602923
$\sum_{m=0}^{\infty}\sum_{n=0}^{\infty} \lvert h_{m,n}\rvert$	2.24766023
$\sum_{m=0}^{\infty}\sum_{n=0}^{\infty} d_{m,n}^2$	1.12468584

Fig. 7. Original "Yogourt" image.

Fig. 8. "Yogourt" processed by filter #2 with scaling factor being 2.5.

Fig. 9. "Yogourt" processed by filter #2 with scaling factor being 3.

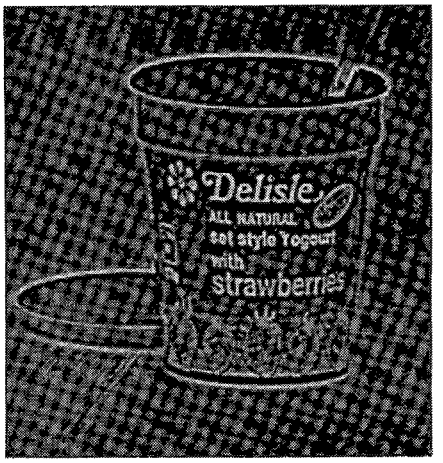


Fig. 10. "Yogourt" processed by filter #3 with scaling factor being 1.5.

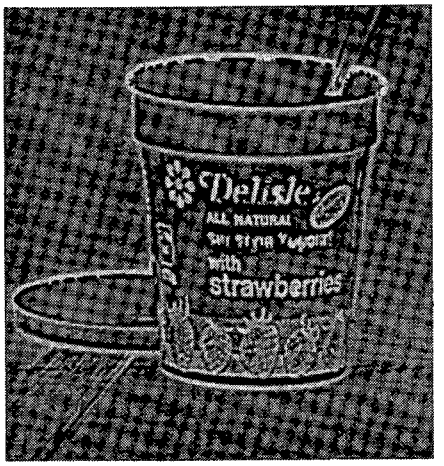


Fig. 11. "Yogourt" processed by filter #3 with scaling factor being 2.

TABLE V
SPECIFICATIONS OF FILTER #3

Coefficients of the Numerator	Coefficients of the Denominator
$a_{0,0} = 0.1$	$b_{0,0} = 1$
$a_{0,1} = 0.230133$	$b_{0,1} = -0.337625 \times 10^{-1}$
$a_{0,2} = 0.1$	$b_{0,2} = 0.134809 \times 10^{-1}$
$a_{1,0} = 0.230133$	$b_{1,0} = -0.337625 \times 10^{-1}$
$a_{1,1} = -1.31453$	$b_{1,1} = 0.1139973932 \times 10^{-2}$
$a_{1,2} = 0.230133$	$b_{1,2} = -0.4551623672 \times 10^{-3}$
$a_{2,0} = 0.1$	$b_{2,0} = 0.134809 \times 10^{-1}$
$a_{2,1} = 0.230133$	$b_{2,1} = -0.4551623672 \times 10^{-3}$
$a_{2,2} = 0.1$	$b_{2,2} = 0.1817346648 \times 10^{-3}$
$\sum_{m=0}^{\infty} \sum_{n=0}^{\infty} h_{m,n}^2$	1.91176681
$\sum_{m=0}^{\infty} \sum_{n=0}^{\infty} \lvert h_{m,n} \rvert$	2.62503134
$\sum_{m=0}^{\infty} \sum_{n=0}^{\infty} d_{m,n}^2$	1.00248344

for image processing is filter #2 (which is one of the filters used in Section IV for simulation) and the second one is filter #3 drawn from [21] (with specifications shown in Table V). Filter #2 is a high-emphasis filter, whereas filter #3 is a high-pass filter. Input images are of size 256×256 pixels with a gray level resolution of 256 levels. The pixel values are from 0 to 255. In order that the input image can be processed by the filter, the input values must lie between -1 and $+1$. Using fixed point two's complement code and 8 bits of accuracy, the pixel value 0 is mapped to -1 while 255 is mapped to $1 - 2^{-7}$.

Fig. 7 shows the original image "Yogourt." Figs. 8–11 are the filtered images processed with different scaling

TABLE VI
SIGNAL-TO-NOISE RATIO OF THE "YOGOURT" IMAGE AFTER FILTERING WITH FILTER #2 USING DIFFERENT SCALE FACTORS

Filter Implementation	Scale Factor	Overflow (Yes/No)	# of pixels with value ≥ 1	# of pixels with value < -1	Signal/Noise (dB)
Stored Product Method	1	No	—	—	32.3382
	2	No	—	—	38.3703
	2.5	Yes	43	60	18.3824
	3	Yes	339	2345	4.1492

TABLE VII
SIGNAL-TO-NOISE RATIO OF THE "YOGOURT" IMAGE AFTER FILTERING WITH FILTER #3 USING DIFFERENT SCALE FACTORS

Filter Implementation	Scale Factor	Overflow (Yes/No)	# of pixels with value ≥ 1	# of pixels with value < -1	Signal/Noise (dB)
Stored Product Method	1	No	—	—	32.3096
	1.5	Yes	10	8	16.8932
	2	Yes	325	223	3.4184

factors. When the scaling factor is too low, the output dynamic range cannot be fully utilized. However, when the scaling factor is too high, overflows occur, as we can obviously notice from Figs. 9 and 11. Although the scaling factor from (26) does not result in any overflows both the signal-to-noise and the visual effect of the processed image are not very good. In order that the full output dynamic range can be utilized, a suitable scaling factor should be used for the scaling of the filter transfer function. There are no general rules for choosing the new scaling factor as this factor is image and filter dependent. Nevertheless, experiments with the filtering of these real images have shown that the scaling factor should be about 3 to 4 times the value given by (26) for the two filters mentioned in order that both the subjective (visual) and objective (signal-to-noise ratio) measurements of performance can be improved. The signal-to-noise here is defined as the ratio of the variance of the ideal filter output signal to the variance of the output error.

The objective measurement of the filtering of the image in Fig. 7 using different scaling factors are shown in Tables VI and VII. From the tables, we can conclude that whenever overflows have occurred, the signal-to-noise ratio is no longer a good indication of the quality of the processed image. The visual effect is still very good even though there are some overflows. Most of the overflows occur when there is a uniform background.

VI. CONCLUSIONS

This paper has considered a new 2-D filter structure which results in a very high operating speed for a 2-D recursive digital filter. The throughput rate is independent of the filter order. The modification made is minimal to the extent that the output sequence of the filter is only delayed by one pixel when compared with the original one. By storing the products in programmable read only memories (PROM's), multipliers can be eliminated completely. This filter configuration provides an economical way of implementing a filter that does not require varying filter coefficients. High-data throughput rate (i.e., 8 MHz) has been shown to be feasible. A simple hardware implementation requiring standard TTL and MOS devices and two clock signals, one being the delayed version of the other, to drive the two different shift registers has been outlined. The proposed hardware is characterized by a regular structure, which consists of identical building blocks.

The noise properties of the new filter implemented by the *stored product* method have been studied. Expressions for estimating the mean and the variance of the noise at the filter output have been derived. Simulation results have been obtained and they agreed very well with the theoretical results.

Finally, the problem of the scaling of the filter transfer function was investigated. Results from the processing of images using a high-pass filter and a high-emphasis filter have shown that a better visual effect of the processed images can be obtained if the scaling factor is three to four times the value given by

$$\rho = \frac{1}{\sum_{k=0}^{\infty}\sum_{l=0}^{\infty}|h_{k,l}|}.$$

ACKNOWLEDGMENT

K. M. Ty would like to express his sincere gratitude to Prof. A. N. Venetsanopoulos for his invaluable advice and guidance.

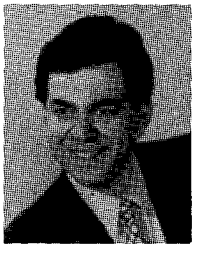
REFERENCES

[1] A. N. Venetsanopoulos and V. Cappellini, "Real-time image processing," in *Multidimensional Systems: Techniques and Applications*, S. G. Tzafestas, Ed. New York: Marcel Dekker, 1986, pp. 345–399.
[2] A. Peled and B. Liu, "A new hardware realization of digital filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-22, no. 6, pp. 456–462, Dec. 1974.
[3] H. Jaggernauth and A. N. Venetsanopoulos, "Distributed arithmetic implementation of two-dimensional filters," in *Proc. IEEE Canadian Communication and Energy Conf.*, Oct. 1982, pp. 407–410.
[4] H. Jaggernauth and A. N. Venetsanopoulos, "Real-time image processing through distributed arithmetic," in *Proc. IEEE Int. Symp. Circuits Syst.* (Newport Beach, CA), May 1–4, 1983, pp. 394–397.
[5] E. Lüder, "Increased speed in digital filters without multipliers," *Arch. Elek. Übertragung.*, pp. 345–348, 1982.
[6] T. Tjahjadi and W. Steenaart, "On the accuracy of ROM stored square multipliers," *IEEE Trans. Circuits Syst.*, vol. CAS-29, pp. 441–447, July 1982.

[7] N. S. Szabo and R. I. Tanaka, *Residue Arithmetic and its Applications to Computer Technology*. New York: McGraw-Hill, 1967.
[8] G. L. Sicuranza, "Memory-oriented realizations of 2-D digital filter," in *Proc. Sixth Eur. Conf. Circuit Theory and Design* (Stuttgart, Federal Republic of Germany), 1983, pp. 447–449.
[9] O. Monkewich and W. Steenaart, "Companding for digital filters," in *Proc. IEEE ISCAS*, 1975, pp. 68–71.
[10] O. Monkewich and W. Steenaart, "Stored product digital filtering with non-linear quantization," in *Proc. IEEE ISCAS*, 1976, pp. 157–160.
[11] D. Dubois and W. Steenaart, "High speed stored product recursive digital filter," *IEEE Trans. Circuits Syst.*, vol. CAS-29, pp. 390–393, June 1982.
[12] K. M. Ty, "Two-dimensional digital filters with minimum cycle time," M. A. Sc. thesis, Dept. Elec. Eng., Univ. Toronto, Toronto, Ontario, Canada, Jan. 1985.
[13] K. M. Ty and A. N. Venetsanopoulos, "Two-dimensional digital filters with minimum cycle time," in *Proc. ICASSP* (Tampa, FL), Mar. 26–29, 1985, vol. 4, pp. 1527–1530.
[14] K. M. Ty and A. N. Venetsanopoulos, "A high speed two-dimensional digital filter structure," in *Proc. of 28th Midwest Symp. Circuits Syst.* (Louisville, KY), Aug. 19–20, 1985, pp. 441–444.
[15] V. A. Dyck, J. D. Lawson, and J. A. Smith, *Introduction to Computing*. Reston, Virginia: Reston Publishing 1979.
[16] A. V. Oppenheim and R. W. Schafer, *Digital Signal Processing*. Englewood Cliff, NJ: Prentice-Hall, 1975.
[17] *The TTL Data Book for Design Engineers*, Texas Instrument Incorporated, 2nd ed., 1981.
[18] *Advanced Micro Devices Condensed Catalog*, Advanced Micro Devices, Inc., 1981.
[19] A. N. Venetsanopoulos, B. G. Mertzios, and S. H. Mneney, "Effects of finite precision in two-dimensional recursive digital filters," *Int. J. Electron.*, vol. 58, no. 1, pp. 159–174, 1985.
[20] E. L Hall, "A comparison of computations for spatial frequency filtering," *Proc. IEEE*, vol. 60, pp. 887–890, July 1972.
[21] B. George, "Design of a 2-D recursive digital filter on the basis of quadrantal symmetry," Master thesis, Dept. Elec. Eng., Univ. Toronto, 1981.

Kich M. Ty (S'85) received the B.A.Sc. (with honors) and M.A.Sc. degrees in electrical engineering from the University of Toronto, Ont., Canada, in 1982 and 1985, respectively. He is now working towards his Ph.D. degree at the same university. He was awarded the J. E. Reid Memorial Prize (communications) from the University of Toronto in 1982, postgraduate fellowships from the same university in 1982, 1983, and 1985, the Connaught Scholarship from the same university in 1984, and Mary. H. Beatty Fellowship in 1986.

His interests include digital signal processing, image processing and analysis, digital video, high-speed processing architecture, and digital communications.



Anastasios N. Venetsanopoulos (S'66–M'69–SM'79) received the B.S. degree in electrical and mechanical engineering from the National Technical University of Athens, Greece (1965), and the M.S. (1966), the M. Phil. (1968), and the Ph.D. (1969) degrees in electrical engineering, all from Yale University. He joined the University of Toronto, Canada, in September 1968, where he is now Professor and Chairman of the Communications Group, Department of Electrical Engineering. He held visiting posts at NTU, the Federal University of Rio de Janeiro, and the University of Florence. He was on research leave at the University of Grenoble, the Imperial College of Science and Technology, and was Adjunct Professor at Concordia University (1981–84).

He has been lecturer of numerous short courses to industry and continuing education programs; contributor to eight books and over 200 papers in digital communications, digital filters, and image processing, and consultant to several organizations. He served as the Assistant Editor (1979–80) and Editor (1981–83) of the *Canadian Electrical Engineering Journal*, the President of the Canadian Society for Electrical Engineering, and Vice-President of the Engineering Institute of Canada (1983–86). He was Fulbright Scholar, an A. F. Schmitt Scholar, and recipient of the J. Vakis Award. He is a member of the New York Academy of Sciences, Sigma Xi, and the Technical Chamber of Greece, a Fellow of the Engineering Institute of Canada, and is a registered Professional Engineer in Ontario and Greece.

Dr. Venetsanopoulos was Program Chairman of the International Communications Conference (ICC'78) and ICC'86. He served as Chairman of the Central Canada Council of IEEE, is presently Associate Editor for Digital Signal Processing of the IEEE TRANSACTIONS ON CIRCUIT AND SYSTEMS, and will be the Guest Editor of the special issue of the same TRANSACTIONS on Digital Image Processing (November 1987).