Nonlinear Filtering by Threshold Decomposition

Jean-Hsang Lin, Nirwan Ansari, *Senior Member, IEEE,* and Jinhui Li

***Abstract*— A new threshold decomposition architecture is introduced to implement stack filters. The architecture is also generalized to a new class of nonlinear filters known as *threshold decomposition* (TD) filters which are shown to be equivalent to the class of Ll-filters under certain conditions. Another new class of filters known as linear and order-statistic (LOS) filters result from the intersection of the class of TD and Ll-filters. Performance comparison among several filters are then presented. It was found that TD is compatible with Ll, LOS, and linear filters in suppressing Gaussian noise, and is superior in suppressing salt-and-pepper noise. LOS filters, however, provide a better compromise in performance and complexity.**

***Index Terms*— L-filters, Ll-filters, linear and order-statistic filters, nonlinear filters, stack filters, threshold decomposition.**

I. INTRODUCTION

LINEAR filters are optimal in eliminating additive white Gaussian noise (AWGN), but in practice, the noise in a channel through which a signal is transmitted is not AWGN; it is not stationary, and it may have unknown characteristics. Therefore, a number of nonlinear filters have been proposed to suppress non-AWGN noise [1]–[5].

Stack filters [1], [6], [7] are a class of sliding-window nonlinear filters characterized by two properties: the threshold decomposition property and the stacking property. They are effective in suppressing impulsive noise, and allow an efficient VLSI implementation. Replacing positive Boolean functions in stack filter by linear operators results in a new class of filters known as *threshold decomposition* (TD) filters, which are more analytically tractable.

Ll-filters [2] are another type of nonlinear filters that generalize the order statistic filters (L-filters) [8], [9] and the nonrecursive linear filters (FIR). Ll-filters are also effective in recovering signals from non-Gaussian noise, and capable of preserving details.

Though the structure of TD filters and Ll-filters are quite different, they still form a common subset—a new type of filters: linear and order-statistic (LOS) filters.

Manuscript received October 7, 1997; revised August 4, 1998. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Henri Maitre.

J.-H. Lin is with the Computer Communications Laboratory, Industrial Technology Research Institute, Hsinchu, Taiwan 310, R.O.C.

N. Ansari is with the Information Engineering Department, Chinese University of Hong Kong, Shatin, Hong Kong (e-mail: nansari@ie.cuhk.edu.hk), on leave from the Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ 07102 USA (e-mail: ang@njit.edu).

J. Li is with the New Jersey Center for Wireless Telecommunications, Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ 07102 USA.

Publisher Item Identifier S 1057-7149(99)05115-5.

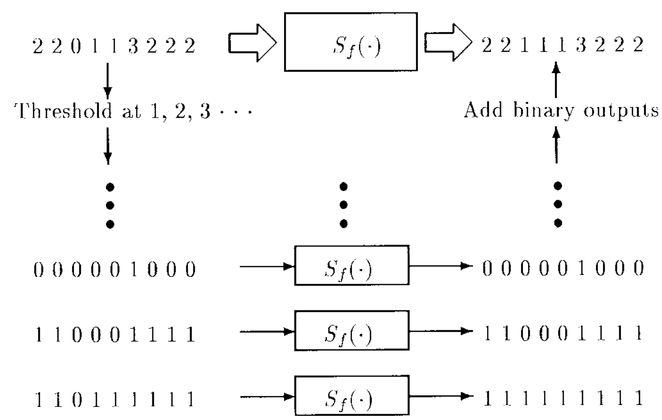


Fig. 1. Stack filter.

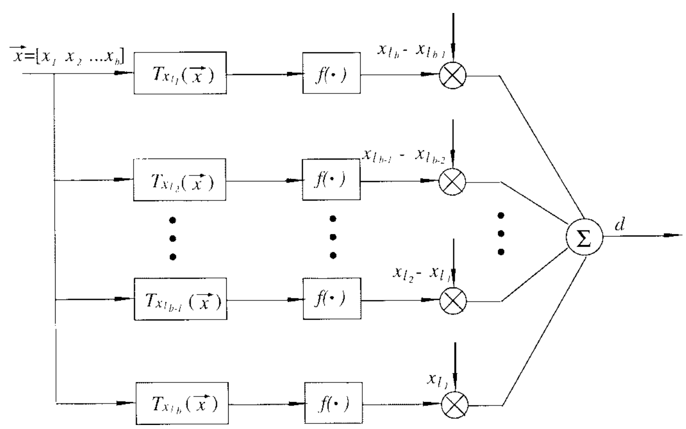


Fig. 2. The new architecture.

II. BACKGROUND

Stack filters is a class of sliding window nonlinear digital filters. Any stack filter can be implemented by the threshold decomposition architecture shown in Fig. 1.

Assume the input $x(n)$ can take on discrete values of $0, 1, \cdots, M-1$, and denote the b samples in the window at time n as

$$\vec{x} = [x_1 \;\; x_2 \;\; \cdots \;\; x_b]. \tag{1}$$

The stack filter's output at time n is

$$S(\vec{x}) = \sum_{k=1}^{M-1} f(T_k(\vec{x})) \tag{2}$$

where T_k denotes the thresholding operation. $T_k(\vec{x})$ is a vector with the same size as $\vec{x}$. $T_k(x_i)$, the ith element of $T_k(\vec{x})$, is one if $x_i \geq k$, and zero, otherwise. f is the positive Boolean function on each level. The index n is omitted for convenience.

(a) (b) (c) (d)

Fig. 3. Experimental results. (a) Original Lena image. (b) Original woman1 image. (c) and (d) Images of (a) and (b) corrupted by Gaussian noise.

In the threshold decomposition architecture, the input signal is decomposed via thresholding into a set of $M-1$ binary signals, and Boolean operation is applied to each of the threshold signals in parallel via Boolean table look-up. The output is the sum of the filtered signals on each level.

It was observed [10] that there are at most b different threshold signals among the $M-1$ threshold signals $T_k(\vec{x})$, $k=1, 2, \cdots, M-1$. These different binary signals can be denoted as $T_{x_{l_i}}(\vec{x})$, $i = 1, 2, \cdots, b$, where l_i denotes the spatial index of the ith rank sample in the window. In other words, x_{l_i} denotes the ith rank sample. Here, a sample of smaller value is given a smaller rank.

By combining repetitive threshold levels, a new architecture for implementing stack filters is introduced as shown in Fig. 2.

$$S(\vec{x}) = \sum_{i=1}^{b} f\Big(T_{x_{l_i}}(\vec{x})\Big)\big(x_{l_i} - x_{l_{i-1}}\big), \qquad x_{l_0} = 0. \quad (3)$$

Note that the number of threshold levels is reduced to b in the new architecture, but extra ranking operation is needed. One distinctive feature of this new architecture is that the threshold decomposition is data-dependent. It leads to the following desirable properties.

Property 2.1: A discrete shift at the input results in a discrete shift at the output. For example, for any integer α,

$$S(\vec{x} + \alpha\vec{1}) = f(\vec{1})\alpha + S(\vec{x}). \quad (4)$$

If we impose $f(\vec{1}) = 1$, then the filtering operation is shift-invariant, where $\vec{1}$ is a vector of size b with each element equal to one.

Property 2.2: The filtering operation is invariant to discrete scale change of the input, i.e., for any integer β

$$S(\beta\vec{x}) = \beta S(\vec{x}). \quad (5)$$

Note that the above properties hold for general filtering operation on each level.

III. TD Filters

In the sequel, we assume different linear operators are used on each level in the new architecture, and the resulting class of filters will be referred to as TD filters. Let the coefficients of the linear operators on the ith level in the new architecture

Fig. 3. (*Continued.*) Experimental results. (e) and (f) are (c) and (d) filtered by a 3 × 3 linear filter that is configured from (a) and (c). (g) and (h) are images of (c) and (d) filtered by a 3 × 3 TD filter that is configured from (a) and (c).

be denoted as $\vec{w}_i$, then the output of the TD filter is

$$S(\vec{x}) = \sum_{i=1}^{b} \vec{w}_i T_{x_{l_i}}(\vec{x})\delta_i \tag{6}$$

where $\vec{w}_i T_{x_{l_i}}(\vec{x})$ denotes the inner product between vector $\vec{w}_i$ and vector $T_{x_{l_i}}(\vec{x})$, and $\delta_i = x_{l_i} - x_{l_{i-1}}$. In general, a TD filter has b^2 coefficients.

Property 3.1: An FIR is a TD filter.

Proof: Any FIR can be regarded as a TD filter which employs the same operator on each level.

Let $x_{i,j} = T_{x_{l_i}}(x_j)$, and denote $w_{i,j}$ as the jth entry of $\vec{w}_i$. Equation (6) can be rewritten as

$$\begin{aligned} S(\vec{x}) &= \sum_{i=1}^{b} \sum_{j=1}^{b} x_{i,j} w_{i,j} \delta_i &(7)\\ &= \sum_{i=1}^{b} \delta_i \sum_{j=1}^{b} x_{i,l_j} w_{i,l_j}. &(8) \end{aligned}$$

Since

$$x_{i,l_j}\delta_i = \begin{cases} \delta_i & \text{if } j \geq i \\ 0 & \text{else} \end{cases} \tag{9}$$

we have

$$S(\vec{x}) = \sum_{i=1}^{b} \delta_i \sum_{j=i}^{b} w_{i,l_j}. \tag{10}$$

Furthermore

$$\begin{aligned} S(\vec{x}) &= \sum_{i=1}^{b} \left(x_{l_i} - x_{l_{i-1}}\right) \sum_{j=i}^{b} w_{i,l_j} \; (x_{l_0} = 0) \\ &= x_{l_b} w_{b,l_b} + x_{l_{b-1}} \left(\sum_{j=b-1}^{b} w_{b-1,l_j} - w_{b,l_b} \right) \\ &\quad + x_{l_{b-2}} \left(\sum_{j=b-2}^{b} w_{b-2,l_j} - \sum_{j=b-1}^{b} w_{b-1,l_j} \right) \\ &\quad + \cdots + x_{l_1} \left(\sum_{j=1}^{b} w_{1,l_j} - \sum_{j=2}^{b} w_{2,l_j} \right) \\ &= x_{l_b} w_{b,l_b} + \sum_{i=1}^{b-1} x_{l_i} \left(\sum_{j=i}^{b} w_{i,l_j} - \sum_{j=i+1}^{b} w_{i+1,l_j} \right). \end{aligned} \tag{11}$$

(i) (j)

(k) (l)

Fig. 3. (*Continued.*) Experimental results. (i) and (j) are images of (c) and (d) filtered by a 3 × 3 Ll-filter that is configured from (a) and (c). (k) and (l) are images of (c) and (d) filtered by a 3 × 3 LOS filter which is configured from (a) and (c).

The above equation indicates that TD filtering is a linear operation where the weights to a sample depends on both its rank and its spatial location. These operations turn out to be similar to Ll filtering [2].

IV. Relation to Ll-Filters

The output of an Ll-filter can be expressed as

$$L(\vec{x}) = \sum_{i=1}^{b} x_{l_i} v_{i,\,l_i} \tag{12}$$

where $v_{i,\,l_i}$ is the weight to the ith rank sample at spatial location l_i. The motivation behind the development of Ll filtering is to enhance the impulsive noise suppression capability of linear filters. The gain in performance is derived from utilizing rank information of the samples in the window.

Even though TD filters and Ll-filters are similar, they are equivalent only when the coefficients satisfy a set of conditions. These conditions are established below.

Property 4.1: An Ll-filter is a TD filter iff its coefficients satisfy the following conditions:

$$v_{i-1,\,j+1} - v_{i-1,\,j} = v_{i,\,j+1} - v_{i,\,j} \tag{13}$$

or equivalently

$$v_{i-1,\,j} - v_{i,\,j} = v_{i-1,\,j+1} - v_{i,\,j+1} \tag{14}$$

for $j = 1, 2, \cdots, b-1$ and $i = 2, 3, \cdots, b$.

Proof: Equating (11) and (12), and by successive substitutions, we have

$$w_{b,\,l_b} = v_{b,\,l_b} \tag{15}$$

$$\sum_{j=b-1}^{b} w_{b-1,\,l_j} = v_{b-1,\,l_{b-1}} + v_{b,\,l_b} \tag{16}$$

$$\sum_{j=b-2}^{b} w_{b-2,\,l_j} = v_{b-2,\,l_{b-2}} + v_{b-1,\,l_{b-1}} + v_{b,\,l_b} \tag{17}$$

(m)

(n)

(o)

(p)

Fig. 3. (*Continued.*) Experimental results. (m) and (n) are images of (c) and (d) filtered by a 3 × 3 median filter. (o) and (p) are images of (a) and (b) corrupted by salt-and-pepper noise.

$$\vdots$$

$$\sum_{j=i}^{b} w_{i,\,l_j} = v_{i,\,l_i} + v_{i+1,\,l_{i+1}} + \cdots + v_{b,\,l_b} \tag{18}$$

$$\vdots$$

$$\sum_{j=2}^{b} w_{2,\,l_j} = v_{2,\,l_2} + v_{3,\,l_3} + \cdots + v_{b,\,l_b} \tag{19}$$

$$\sum_{j=1}^{b} w_{1,\,l_j} = v_{1,\,l_1} + v_{2,\,l_2} + \cdots + v_{b,\,l_b}. \tag{20}$$

Note that in the above equations, the left hand side remains the same for any permutation of $(l_i,\, l_{i+1},\, \cdots,\, l_b)$. Hence, there exists a solution $w_{i,\,j}'^{s}$ for a given $v_{i,\,j}'^{s}$ only if the right hand side (RHS) remains constant for any permutation of $(l_i,\, l_{i+1},\, \cdots,\, l_b)$.

Given the condition in (18), a swap of any two indices of $(l_i,\, l_{i+1},\, \cdots,\, l_b)$ does not change the value of the RHS. Since any two permutations can be related via successive swaps of two indices, the RHS of (18) is constant for any permutation of $(l_i,\, l_{i+1},\, \cdots,\, l_b)$. This establishes (13) as a sufficient condition.

In (18), let the subset of indices $(l_{i+2},\, l_{i+3},\, \cdots,\, l_b)$ be fixed, then (13) is necessary to keep the RHS constant when $(l_i,\, l_{i+1})$ is swapped. This establishes (13) as a necessary condition. Hence, we have shown that (13) is a necessary and sufficient condition for an Ll-filter to be a TD filter.

Property 4.2: A TD filter is an Ll-filter iff its coefficients satisfy the following conditions:

$$w_{i,\,j} - w_{i+1,\,j} = w_{i,\,j+1} - w_{i+1,\,j+1} \tag{21}$$

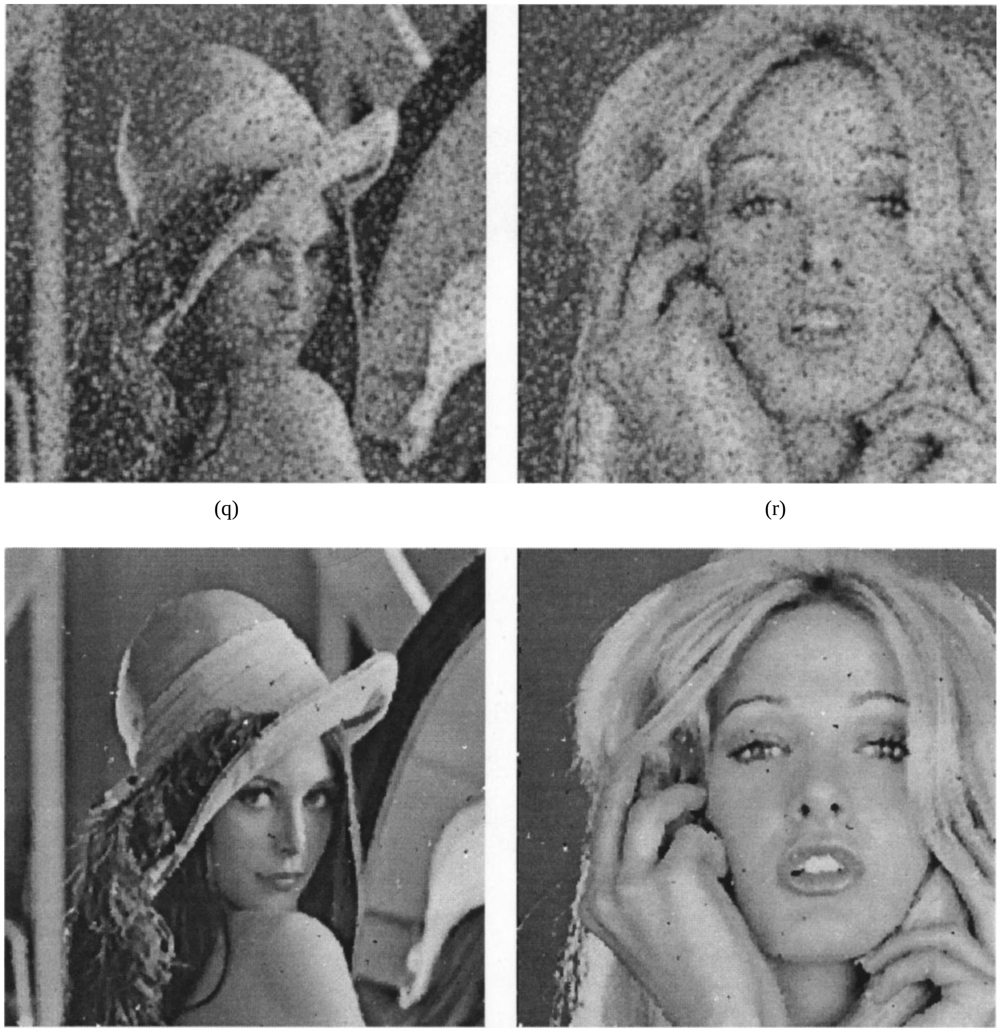
or

$$w_{i,\,j} - w_{i,\,j+1} = w_{i+1,\,j} - w_{i+1,\,j+1} \tag{22}$$

for $i = 1,\, 2,\, \cdots,\, b-1$, and $j = 1,\, 2,\, \cdots,\, b-1$.

Proof: The proof is similar to the above, and is thus omitted.

From the above two properties, it can be concluded that only a small subclass of Ll-filters and TD filters are equivalent. In this subclass, each filter possesses $2b-1$ independent coefficients. Following the previous notation, denote the coefficients



(q) (r)

(s) (t)

Fig. 3. (*Continued.*) Experimental results. (q) and (r) are images of (o) and (p) filtered by a 3 × 3 linear filter that is configured from (a) and (o). (s) and (t) are images of (o) and (p) filtered by a 3 × 3 TD filter that is configured from (a) and (o).

on the ith level as $\vec{w}_i$, then

$$\vec{w}_i = \vec{w} + \alpha_i \vec{1} \tag{23}$$

where $\vec{w}$ contains b independent coefficients. Since there are only $2b-1$ independent coefficients, one of α_i can be set to zero. For convenience, we do not impose this condition. Substituting (23) into (6)

$$S(\vec{x}) = \sum_{i=1}^{b} (\vec{w} + \alpha_i \vec{1}) T_{x_{l_i}}(\vec{x}) \delta_i \tag{24}$$

$$= \vec{w} \sum_{i=1}^{b} T_{x_{l_i}}(\vec{x}) \delta_i + \sum_{i=1}^{b} (b+1-i) \alpha_i (x_{l_i} - x_{l_{i-1}}) \tag{25}$$

$$= \vec{w}\vec{x} + \vec{v}\vec{x}_s \tag{26}$$

where $\vec{x}_s$ denotes the sorted $\vec{x}$. Hence, any filter in the subclass can be implemented as a linear filter and an order-statistic filter interconnected in parallel. For convenience, they will be referred to as LOS filters.

Equations (24)–(26) demonstrate the following two properties of LOS filters.

Property 4.3: An FIR is a LOS filter.

Property 4.4: An order statistic filter (L-filter) is a LOS filter.

The following property results from Property 4.4 immediately.

Property 4.5: An order statistic filter (L-filter) is a TD filter.

V. Performance Comparison

According to (7), the output of TD filter $S(\vec{x})$ is a linear combination of $x_{i,j}\delta_i$ ($i = 1, 2, \cdots, b$ and $j = 1, 2, \cdots, b$). Just like linear filters, the optimal TD filter in this case under the least mean squares (LMS) criterion satisfies the following equation:

$$E[\vec{u}\vec{u}^t]\vec{w} = E[\vec{u}d] \tag{27}$$

where $\vec{u} = [x_{1,1}\delta_1, x_{1,2}\delta_1, \cdots, x_{1,b}\delta_1, x_{2,1}\delta_2, \cdots, x_{b,b}\delta_b]^t$, $\vec{w}$ is the weight vector having the same dimension as $\vec{u}$, and d is the desired output.

Fig. 3. (*Continued.*) Experimental results. (u) and (v) are images of (o) and (p) filtered by a 3×3 Ll-filter which is configured from (a) and (o). (w) and (x) are images of (o) and (p) filtered by a 3×3 LOS filter which is configured from (a) and (o). (y) and (z) are images of (o) and (p) filtered by a 3×3 median filter.

In practice, the expectation is replaced by the averaging operator, and (27) is simplified as

$$\mathbf{R}\vec{w} = \vec{P} \tag{28}$$

where $\mathbf{R} = \overline{(\vec{u}\vec{u}^t)}$ and $\vec{P} = \overline{(\vec{u}d)}$, and $\overline{(\cdot)}$ is the averaging operator.

When matrix $\mathbf{R}$ is nonsingular, $\vec{w} = \mathbf{R}^{-1}\vec{P}$. Otherwise, the number of solutions will be infinite. Any solution can be

TABLE I
MAE AND RMSE OF NOISY (GAUSSIAN NOISE) IMAGES AND OUTPUT OF VARIOUS FILTERS

	lena		woman1	
	MAE	RMSE	MAE	RMSE
Noisy image	21.90	27.32	22.37	28.04
3 × 3 Linear filtering	9.47	12.34	10.59	14.17
3 × 3 TD filtering	9.42	12.24	10.57	14.15
3 × 3 Ll filtering	9.38	12.18	10.50	14.01
3 × 3 LOS filtering	9.44	12.28	10.57	14.14
3 × 3 L-filtering	9.57	12.68	10.84	14.87
3 × 3 Median filtering	10.95	14.26	12.03	16.32

TABLE II
MAE AND RMSE OF NOISY (SALT-AND-PEPPER NOISE) IMAGES AND OUTPUT OF VARIOUS FILTERS

	lena		woman1	
	MAE	RMSE	MAE	RMSE
Noisy image	20.43	56.37	20.50	54.06
3 × 3 Linear filtering	16.94	22.60	17.64	23.29
3 × 3 TD filtering	4.05	9.08	5.67	11.95
3 × 3 Ll filtering	4.41	9.24	5.64	11.95
3 × 3 LOS filtering	4.67	9.64	6.39	12.71
3 × 3 L-filtering	4.61	9.68	6.38	12.81
3 × 3 Median filtering	4.35	9.87	6.17	13.10

expressed as

$$\vec{w} = \vec{w}_P + \vec{w}_N \tag{29}$$

where $\vec{w}_P$ satisfies (28), and $\vec{w}_N$ belongs to the null space, i.e.,

$$\mathbf{R}\vec{w}_N = 0. \tag{30}$$

Therefore, each solution yields the same mean square error (MSE):

$$\text{MSE} = -\vec{w}_P^t \mathbf{R} \vec{w}_P + \overline{d^2}. \tag{31}$$

In our experiment, two types of noise are used: Gaussian noise and salt-and-pepper noise. To configure a filter, the original lena image is referenced as the desired output, and its noisy version is employed as the input. That is, given the two images, the weights of the optimal TD filter is obtained by solving (28). This filter is then used to filter the noisy Lena image and the woman1 image corrupted by the same type of noise.

By the same token, the optimal Ll-filters, LOS filters, and linear filters under the LMS criterion also satisfy (28), except that the definition of $\vec{u}$ and the size of the vectors are different. In order to configure Ll-filters, $\vec{u}$ is defined as $\vec{u} = [x_{l_1}\delta(1-l_1), x_{l_1}\delta(2-l_1), \cdots, x_{l_1}\delta(b-l_1), x_{l_2}\delta(1-l_2), \cdots, x_{l_b}\delta(b-l_b)]^t$, where $\delta(n)$ is one if $n = 0$, and zero, otherwise; $\vec{u} = [x_1, x_2, \cdots, x_b, x_{l_2} - x_{l_1}, \cdots, x_{l_b} - x_{l_{b-1}}]^t$ for LOS filters, $\vec{u} = [x_{l_1}, x_{l_2}, \cdots, x_{l_b}]^t$ for L filters, and $\vec{u} = [x_1, x_2, \cdots, x_b]^t$ for linear filters.

Fig. 3 shows the original images, noisy images, and filtered images of lena and woman1, respectively. Mean absolute error (MAE) and root mean square error (RMSE) of the noisy images and filtered images are tabulated in Tables I and II. For comparison purposes, results of median filtering are also included in the figures and tables. The resolution of all images is 256 × 256, 8 b/pixel. Variances of the original Lena and woman1 images are 2734 and 1811, respectively. The window size used in our experiments is 3 × 3.

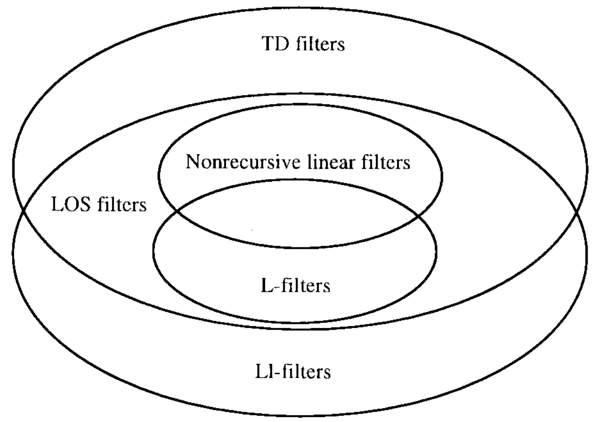


Fig. 4. Relationship among different classes of filters.

VI. CONCLUSIONS

The relationship among several types of filters is illustrated in Fig. 4. The intersection between the TD filters and Ll-filters forms the LOS filters, a simple addition of linear operation and L-filtering. It is evident that LOS filters generalize FIR and L-filters, and thus FIR and L-filters are subsets of TD filters and Ll-filters.

According to Tables I and II, the performance in suppressing Gaussian noise among TD filters, Ll-filters, LOS filters and linear filters are similar. Median filtering performs poorly in suppressing Gaussian noise as expected. Among the filters tested on these images, Ll achieves the best performance in suppressing Gaussian noise while TD suppresses salt-and-pepper noise the best.

With a window size of b, it requires b coefficients to configure a linear filter, b^2 coefficients to configure a TD or Ll-filter, and only $2b - 1$ coefficients to configure a LOS filter. Therefore, LOS filters provide a trade-off between performance and complexity.

REFERENCES

[1] P. D. Wendt, E. J. Coyle, and N. C. Gallagher, Jr., "Stack filters," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-34, pp. 898–911, Aug. 1986.
[2] F. Palmieri and C. G. Boncelet, Jr., "Ll-Filters—A new class of order statistic filters," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. 37, pp. 691–701, May 1989.
[3] Z. Z. Zhang and N. Ansari, "Structure and properties of generalized adaptive neural filters for signal enhancement," *IEEE Trans. Neural Networks,* vol. 7, pp. 857–868, July 1996.
[4] H. Hanek and N. Ansari, "Speeding up the generalized adaptive neural filters," *IEEE Trans. Image Processing,* vol. 5, pp. 705–712, May 1996.
[5] J.-H. Lin and E. J. Coyle, "Minimum mean absolute error estimation over the class of generalized stack filters," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. 38, pp. 663–678 Apr. 1990.
[6] J.-H. Lin, T. M. Sellke, and E. J. Coyle, "Adaptive stack filtering under the mean absolute error criterion," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. 38, pp. 938–954, June 1990.
[7] G. C. Gurski and M. T. Orchard, "Optimal stack filters for pyramidal decomposition," in *Proc. 25th Annual Conf. Information Science and*

Systems, Dept. Elect. Comput. Eng., Johns Hopkins Univ., Baltimore, MD, Mar. 20–22, 1991.

[8] H. A. David, *Order Statistics.* New York: Wiley, 1981.

[9] E. Sarhan and B. G. Greenberg, *Contributions to Order Statistics.* New York: Wiley, 1962.

[10] J.-H. Lin, Y. T. Kim, and G. Soemarwoto, "Nonlinear filtering techniques based on a new threshold decomposition architecture," in *Proc. 26th Ann. Conf. Information Science and Systems*, Princeton, NJ, Mar. 18–20, 1992.

Jean-Hsang Lin received the B.Sc. degree in electronics engineering from the Chung-Yuan Institute of Technology, Chung Li City, Taiwan, R.O.C., in 1977, the M.Sc. degree in electrical engineering from the the University of Texas, Arlington, in 1984, and the Ph.D. degree in electrical engineering from and Purdue University, West Lafayette, IN, in 1989.

During 1981–1982, he was a Design Engineer at GTE Telecommunications Taiwan Ltd., Hsinchu, responsible for digital system design. In 1989, he joined the Department of Electrical Engineering at the University of Delaware, Newark, as an Assistant Professor, where his main research is in nonlinear digital suppression. Since 1996, he has been with CCL/ITRI, Taiwan, pursuing wireless communications research and development. Currently, he is involved in W-CDMA technology evaluation.

Nirwan Ansari (S'78–M'83–SM'94) received the B.S.E.E. degree (summa cum laude) from the New Jersey Institute of Technology (NJIT), Newark, the M.S.E.E. degree from the University of Michigan, Ann Arbor, and the Ph.D. degree from Purdue University, West Lafayette, IN, in 1982, 1983, and 1988, respectively.

He has been a Professor in the Department of Electrical and Computer Engineering, NJIT, since 1997. He visited the Department of Information Engineering of the Chinese University of Hong Kong during the 1998–1999 academic year. He co-authored (with E.S.H. Hou) *Computational Intelligence for Optimization* (Boston, MA: Kluwer, 1997), and co-edited (with B. Yuhas) *Neural Networks in Telecommunications* (Boston, MA: Kluwer, 1994).

Dr. Ansari is a Technical Editor of the IEEE COMMUNICATIONS MAGAZINE and the Chair of the North Jersey Chapter of the IEEE COMSOC that received the 1996 Chapter of the Year Award. He also serves in various IEEE committees. He was the 1998 recipient of an IEEE Region I Award, and the 1998 recipient of the NJIT Excellence Teaching Award in Graduate Instruction.

Jinhui Li received the B.S. degree in physics and the M.S. degree in electrical engineering from Peking University, Beijing, China, in 1991 and 1994, respectively. Currently, he is pursuing the Ph.D. degree in electrical engineering at the New Jersey Institute of Technology, Newark.