Dual Stack Filters and the Modified Difference of Estimates Approach to Edge Detection

Jisang Yoo, Edward J. Coyle, *Senior Member, IEEE,* and Charles A. Bouman, *Member, IEEE*

***Abstract*—The theory of optimal stack filtering has been used in the difference of estimates (DoE) approach to the detection of intensity edges in noisy images. In this paper, the DoE approach is modified by imposing a symmetry condition on the data used to train the two stack filters. Under this condition, the stack filters obtained are duals of each other. Only one filter must therefore be trained; the other is simply its dual. This new technique is called the symmetric difference of estimates (SDoE) approach. The dual stack filters obtained under the SDoE approach are shown to be comparable. This allows the difference of these two filters to be represented by a single equivalent edge operator. This latter result suggests that an edge operator can be found by directly training a (possibly nonpositive) Boolean function to be used on each level of the threshold decomposition architecture. This approach, which is called the threshold Boolean filter (TBF) approach, requires less training time but produces operators that are less robust than those produced by the SDoE approach. This is demonstrated and interpreted via comparisons of results for natural images.**

***Index Terms*— Boolean function, duality, edge detection, robustness, stack filter.**

I. Introduction

An IMPORTANT current emphasis in schemes for the detection of intensity edges is the reliable detection of these edges even when the image has been corrupted by noise. In fact, the structure of the edge detector is often heavily influenced by the type of noise that is expected. As a result, edge detection schemes that work well for one type of noise may perform poorly for other noise types.

A robust edge detection scheme called the difference of estimates (DoE) approach has been developed [1]. In this scheme, the edge detection problem is recast as an estimation problem, and stack filters are used to produce the required estimates. The distributional robustness and detail preserving capability of stack filters allow very effective edge detection, even in the presence of poorly characterized noise processes.

Manuscript received April 5, 1996; revised April 2, 1997. This work was supported in part by Grant 96166-IT1-I2 from the University Fundamental Research Foundation of the Korean Ministry of Information and Communication. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Henri Maitre.

J. Yoo is with the Department of Electronics Engineering, Kwangwoon University, Seoul 139-701, Korea (e-mail: jsyoo@daisy.kwangwoon.ac.kr).

E. J. Coyle and C. A. Bouman are with the School of Electrical Engineering, Purdue University, West Lafayette, IN 47907-0501 USA.

Publisher Item Identifier S 1057-7149(97)08488-1.

In this approach, *two* stack filters are applied to a noisy image to obtain local estimates of the dilated and eroded versions of the noise-free image. The difference between these two estimates is an estimate of the maximum absolute gradient inside the window [1]. An accurate edge map is produced by thresholding this gradient estimate.

In this paper, two new approaches to the design of edge detectors are developed. Both are motivated by the structure of the DoE operator. They are called the symmetric difference of estimates (SDoE) approach and the threshold Boolean filter (TBF) approach.

In the SDoE approach, a symmetry constraint is imposed on the data used to train the two stack filters in the DoE approach. It is satisfied if the training data is the union of the desired training image and its inverse. With this constraint, the two stack filters produced are duals of each other and they produce statistically unbiased estimates. The fact that the two filters are duals of each other implies that only one filter need be trained; the other is obtained as its dual.

It is further shown that the two stack filters produced under the SDoE approach are always comparable. Their difference is then shown to be a filter that possesses the threshold decomposition architecture and property of stack filters even though the Boolean filter used on each level of the architecture is no longer positive. This equivalent edge operator thus falls within the class of nonlinear filters known as threshold Boolean filters [2], [3].

This latter result suggests that we consider edge operators that have the threshold decomposition architecture of stack filters, but in which the Boolean operator on each level is arbitrary; i.e., we no longer require that it obey the stacking property. This Boolean function can be directly designed to estimate the difference of the dilated and eroded versions of the noise-free original image on each threshold level. This approach is called the threshold Boolean function (TBF) approach.

As discussed in [3], it is difficult to determine a Boolean function that produces a TBF that achieves the minimum error under either the mean square error criterion or the mean absolute error criterion. If, however, the error criterion is chosen to be the sum over all threshold levels of the absolute error on each level, this problem can be solved by using minimum mean absolute error (MMAE) stack filter design techniques [4]. Note that the multilevel mean absolute error is always less than or equal to the sum of the mean absolute errors on each threshold level. Equality is achieved if and only if the Boolean function on each level has the stacking property (is positive).

Our strategy is then to minimize this bound of the absolute error between the output of the TBF and the noise-free edge map. The problem of designing an optimal Boolean function is then equivalent to using the MMAE design technique on each threshold level. We can, in fact, use the training algorithm developed in [4] by removing the swap operations that enforce the stacking property. The algorithm runs faster when the stacking property is not enforced, leading to reduced training time.

Because of the reduced training time required in the TBF approach, it would appear to be superior to the two DoE approaches. This apparent advantage is reinforced by the comparisons in Section IV, which demonstrate that the TBF approach often—but not always—yields lower MAE than the DoE operators when they are applied to the images on which they were trained. Unfortunately, the error is often lower only for the training images—the TBF approach produces larger errors, as shown in Section IV via the MAE criterion and judgments of edge-map quality, for the target image. In other words, the TBF approach exhibits less robustness than either of the DoE approaches. One possible explanation for this phenomenon is that the removal of the stacking property leads to filters which are no longer cascades and compositions of rank order operators, whose robustness has been well established in the statistics literature.

There is, thus, a trade-off to be faced when choosing between the TBF approach and the DoE approaches. If it is known that the training image is very similar to the target image, and that the noise used in training is similar to that expected in the target image, then the TBF approach could be used since it offers reduced training time. If the target image may differ significantly from the training image, or if some mismatch in noise is possible between the training noise and the noise that might be encountered, one of the DoE approaches should be used.

This paper is organized as follows. The notation used throughout the paper and a review of the relevant properties of stack filters are provided in Section II. In Section II, we also review the optimal stack filtering algorithm, define the notion of a symmetric image, and discuss duality in the context of stack filters. In the following section, the DoE approach to edge detection is reviewed and the SDoE approach and the TBF approach are introduced. Performance and complexity comparisons between the DoE and TBF approaches are provided in Section IV.

II. Stack Filters

A. Properties of Stack Filters

Stack filters are a class of nonlinear filters that satisfy two properties: the weak superposition property known as the threshold decomposition and the ordering property called the stacking property in [5].

To define these properties and establish the notation used throughout this paper, we must introduce the threshold decomposition of an image. Images will be assumed to take values on a lattice S, with each point in the lattice denoted by $s \in S$. A gray scale image X with pixel values ranging between 0 and M, may be represented as the sum of a series of binary-valued images,

$$X(s) = \sum_{l=1}^{M} x(s, l),$$

where, for each l, the binary image $x(s, l)$ is obtained by thresholding $X(s)$ at level l. Then

$$x(s, l) = \begin{cases} 1, & X(s) \geq l \\ 0, & X(s) < l \end{cases}.$$

For a particular value of s, define $W_X(s)$ to be the vector of pixels in X occurring at the points $s + r$, $r \in W$. If, for example, W is a 2×2 square, then when its upper left corner is located at $s = (s_1, s_2)$

$$W_X((s_1, s_2)) = [X((s_1, s_2)), X((s_1+1, s_2)), X((s_1, s_2+1)), X((s_1+1, s_2+1))].$$

The threshold decomposition of the window W_X, when it is located at the point s in the lattice S, is then,

$$W_X(s) = \sum_{l=1}^{M} w_X(s, l),$$

where $w_X(s, l)$ is the binary window vector at the point s in the binary image $x(s, l)$ obtained by thresholding $X(s)$ at level l.

The threshold decomposition and stacking properties of a stack filter $S_f(\cdot)$ can now be defined. Note that the threshold decomposition property is a weak superposition property.

Definition 2.1—The Threshold Decomposition Property:

$$\begin{aligned} S_f[W_X(s)] &= S_f\left[\sum_{l=1}^{M} w_X(s, l)\right] \\ &= \sum_{l=1}^{M} S_f[w_X(s, l)] \\ &= \sum_{l=1}^{M} f[w_X(s, l)]. \end{aligned}$$

Definition 2.2—The Stacking Property: If the output of f applied to $w_X(s, l)$ is one, then the output of f applied to $w_X(s, k)$ for any $k \leq l$ must also be one's. More formally, for all $k \leq l$

$$f[w_X(s, k)] \geq f[w_X(s, l)]. \tag{1}$$

A Boolean function has this property if and only if it is positive [6]. For the thresholded binary inputs, the operation of a stack filter $S_f(\cdot)$ is the same as the operation of the positive Boolean function $f(\cdot)$. We can thus say that the positive Boolean function f defines the stack filter $S_f(\cdot)$. If, for example, the stack filter is the median filter, then the positive Boolean function that defines it is the majority logic operator.

Any Boolean function f of N variables can be represented by a length 2^N decision vector D, $D = (d_1, d_2, \cdots, d_{2^N})$, where $d_i = f(\alpha_i)$ and α_i is one of the binary vectors with N

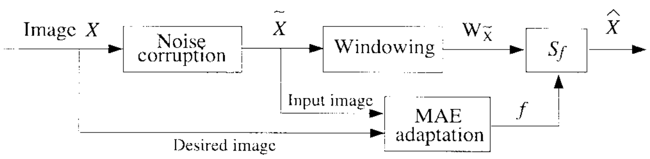


Fig. 1. Block diagram of the adaptive stack filtering algorithm.

elements. Therefore, the problem of finding a positive Boolean function that defines an optimal stack filter is equivalent to finding the decision vector for this function.

Since a stack filter is completely specified by its positive Boolean function, a dual stack filter can be defined by the dual of this positive Boolean function.

Definition 2.3: The dual of a Boolean function $f(a)$, denoted by $f^{\mathrm{d}}(a)$, is defined as

$$f^{\mathrm{d}}(a) = \overline{f}(\overline{a})$$

where $a = (a_1, a_2, \cdots, a_n)$ is a binary input vector and $\overline{f}$ and $\overline{a}$ are the inversion of f and a, respectively.

If a Boolean function f is positive, then its dual f^{d} is also positive. This implies that if a Boolean function defines a stack filter, then its dual also defines a stack filter.

Definition 2.4: The dual of a stack filter S_f is the stack filter defined by f^{d}, the dual of f. This dual stack filter is denoted by S_f^{d}.

For instance, the dual of the max filter is the min filter. The dual of the median filter is the median filter; that is, the median is self-dual.

B. Optimal Stack Filtering

Optimal adaptive stack filtering is one approach to the design of stack filters [4]. Fig. 1 illustrates the procedure for adaptive stack filtering.

For a particular window size, the goal of the optimal filtering problem is to find a stack filter that minimizes the mean absolute error (MAE) between the output of the stack filter and the desired image. If $W_{\tilde{X}}$ and $X(s)$ are jointly stationary random processes, then the cost to be minimized by proper choice of f is

$$\begin{aligned}\mathrm{MAE}_f &= E\{\,|\,S_f[W_{\tilde{X}}(s)] - X(s)\,|\,\} \\ &= \sum_{l=1}^{M} E\{\,|\,f[w_{\tilde{X}}(s, l)] - x(s, l)\,|\,\} \qquad (2)\end{aligned}$$

where X is a desired image and $\tilde{X}$ is the corrupted version of X. The optimal filtering problem is thus equivalent to finding the positive Boolean function f which minimizes (2).

C. Duality

In the DoE approach, two stack filters must be trained. One is trained to estimate the dilated version (maximum) of the noise-free image; the other is trained to estimate the eroded version (minimum) of the noise-free image [1].

Consider a stack filter $S_{f_{\mathrm{d}}}$ which optimally estimates the dilated version of the noise-free image. If the dual of $S_{f_{\mathrm{d}}}$ is optimal for the eroded version of the noise-free image, then we can design the DoE operator with only one training run. This will only occur, though, if the statistics of the training data satisfy a certain symmetry condition.

To define this symmetry condition, let $A_N = \{a_i\colon i = 1, 2, \cdots, 2^N\}$ be the set of 2^N binary vectors of length N. Then for every position s in the lattice and every threshold level l, the binary window vector $w_X(s, l)$ is contained in A_N. For each i, define $O_X(a_i)$ to be the number of occurrences of the binary vector a_i in the image X.

Definition 2.5: An image X is said to be symmetric if and only if $O_X(a_i) = O_X(\overline{a}_i)$ for all i, where $\overline{a}_i$ is the binary vector obtained by complementing each entry of a_i.

Although most images are not symmetric in the above sense, it is not difficult to generate symmetric training images. The image shown in Fig. 2(a) is the 256 × 256 natural image "couple" with 256 gray levels. Its inverse, which is obtained by taking the 256's complement of each of its pixels (replace each pixel value by 256 minus that value), is shown in Fig. 2(b). The union of these two images is a symmetric image. Let T be the lattice defining the location of the pixels in a symmetric image. Then, T has two sublattices, P and Q with $T = P \cup Q$, which correspond to the locations of the pixels in the original image and its 256's complement, respectively.

The optimal stack filtering problem being considered must be generalized in the following fashion in order to discuss the DoE approach to edge detection. Define $Y(s)$ to be the image obtained when the stack filter S_{g} is applied to a noise-free image $X(s)$; that is, $Y(s) = S_{\mathrm{g}}[W_X(s)]$ for each s. Our goal is to find a stack filter $S_{f_{\mathrm{opt}}}$ which achieves the following:

$$\begin{aligned}\mathrm{MAE}_{f_{\mathrm{opt}}} &= \min_f [E\{\,|\,S_f[W_{\tilde{X}}(s)] - Y(s)\,|\,\}] \\ &= \min_f \left[\sum_{l=1}^{M} E\{\,|\,f[w_{\tilde{X}}(s, l)] - y(s, l)\,|\,\}\right] \\ &= \min_f \left[\sum_{l=1}^{M} E\{\,|\,f[w_{\tilde{X}}(s, l)] - g[w_X(s, l)]\,|\,\}\right]\end{aligned}$$

in which $\tilde{X}(s)$ is the noise-corrupted version of the image $X(s)$, and $y(s, l)$ is the binary image obtained by thresholding $Y(s)$ at level l. We then say that $S_{f_{\mathrm{opt}}}$ is an optimal stack filter for estimating $S_{\mathrm{g}}(X)$, or equivalently, that f_{opt} is an optimal positive Boolean function (binary stack filter) for estimating $g(x)$.

Theorem 2.1: If a symmetric image is used to train a stack filter $S_{f_{\mathrm{opt}}}$ which is an optimal stack filter for estimating $S_{\mathrm{g}}(X)$, then $S_{f_{\mathrm{opt}}}^{\mathrm{d}}$ is an optimal stack filter for estimating $S_{\mathrm{g}}^{\mathrm{d}}(X)$.

Proof: Assume that the symmetric image X ($\tilde{X}$) has been obtained as described above—by taking the union of the original image (noise corrupted image) and its 256's complement. Let T be the lattice defining the location of the pixels in X, which has twice as many pixels as the original image. This lattice has two sublattices, P and Q with $T = P \cup Q$, which correspond to the locations of the pixels in the original image and its 256's complement, respectively. Also, each pixel takes on one of the $M + 1$ values from 0 to

(a) (b)

Fig. 2. (a) 256 × 256 natural image "couple" with 256 gray levels and (b) the inverse of (a). These images are used together to obtain symmetric training of filters.

M; the window used by the filter $S_{f_{\mathrm{opt}}}$ is denoted by W; and, the window used by S_{g} is denoted by V.

Since f_{opt} is an optimal estimator for $g(X)$, and since $|h-k| = |\overline{h} - \overline{k}|$ for any two Boolean functions h and k and their inverses $\overline{h}$ and $\overline{k}$, the (minimum) total absolute error between the outputs of f_{opt} and g is

$$\begin{aligned}
&\sum_{s\in T}\sum_{l=1}^{M} |f_{\mathrm{opt}}[w_{\tilde{X}}(s,l)] - g[v_X(s,l)]| \\
&\quad = \sum_{s\in T}\sum_{l=1}^{M} |\overline{f}_{\mathrm{opt}}[w_{\tilde{X}}(s,l)] - \overline{g}[v_X(s,l)]| \\
&\quad = \sum_{s\in P}\sum_{l=1}^{M} |\overline{f}_{\mathrm{opt}}[w_{\tilde{X}}(s,l)] - \overline{g}[v_X(s,l)]| \\
&\qquad + \sum_{s\in Q}\sum_{l=1}^{M} |\overline{f}_{\mathrm{opt}}[w_{\tilde{X}}(s,l)] - \overline{g}[v_X(s,l)]|. \qquad (3)
\end{aligned}$$

Since X and $\tilde{X}$ are symmetric, (3) can be rewritten as follows.

$$\begin{aligned}
&\sum_{s\in Q}\sum_{l=1}^{M} |\overline{f}_{\mathrm{opt}}[\overline{w}_{\tilde{X}}(s,l)] - \overline{g}[\overline{v}_X(s,l)]| \\
&\qquad + \sum_{s\in P}\sum_{l=1}^{M} |\overline{f}_{\mathrm{opt}}[\overline{w}_{\tilde{X}}(s,l)] - \overline{g}[\overline{v}_X(s,l)]| \\
&\quad = \sum_{s\in T}\sum_{l=1}^{M} |\overline{f}_{\mathrm{opt}}[\overline{w}_{\tilde{X}}(s,l)] - \overline{g}[\overline{v}_X(s,l)]| \\
&\quad = \sum_{s\in T}\sum_{l=1}^{M} |f_{\mathrm{opt}}^{\mathrm{d}}[w_{\tilde{X}}(s,l)] - g^{\mathrm{d}}[v_X(s,l)]|.
\end{aligned}$$

Therefore, if f_{opt} gives the optimal estimate of the output of g then its dual, $f_{\mathrm{opt}}^{\mathrm{d}}$, gives the optimal estimate for g^{d}. □

Corollary 2.1: If $S_{f_{\mathrm{opt}}}$ is obtained with symmetric images, the total absolute error between the outputs of $S_{f_{\mathrm{opt}}}$ and S_{g} is the same as the total absolute error between the outputs of $S_{f_{\mathrm{opt}}^{\mathrm{d}}}$ and $S_{\mathrm{g}}^{\mathrm{d}}$.

III. Modified Difference of Estimates Operators

A. Symmetric Difference of Estimates Operator

In the Difference of Estimates approach, two stack filters, $S_{f_{\mathrm{d}}}$ and $S_{f_{\mathrm{e}}}$, are trained to produce local estimates of the dilated noise-free image and the eroded noise-free image, respectively. Let X be a noise-free training image with $M+1$ gray levels. Assume that X is not directly available, so that the noisy image $\tilde{X}$ must be used to determine edge locations.

The DoE operator is defined as

$$\begin{aligned}
\mathrm{DoE}_{\tilde{X}}(s) &= S_{f_{\mathrm{d}}}[W_{\tilde{X}}(s)] - S_{f_{\mathrm{e}}}[W_{\tilde{X}}(s)] \\
&= \sum_{l=1}^{M} \{f_{\mathrm{d}}[w_{\tilde{X}}(s,l)] - f_{\mathrm{e}}[w_{\tilde{X}}(s,l)]\} \qquad (4)
\end{aligned}$$

where, as before, W is the window used for both stack filters. This window W is usually larger than the 2 × 2 window used by the max and min filters to produce the eroded and dilated versions of the noise-free image X [1].

The objective is to choose $S_{f_{\mathrm{d}}}$ and $S_{f_{\mathrm{e}}}$ so that the difference of estimates, $\mathrm{DoE}_{\tilde{X}}$, is a good approximation to the difference, D_X, of the outputs of the 2 × 2 maximum and minimum filters with the noiseless image as input—note that this latter difference is the largest magnitude assumed by the gradient when it is computed in all directions inside the 2 × 2 window. The estimated edge map is produced by thresholding the output of the DoE operator.

Since the difference operator does not have the stacking property, we cannot directly use the methods of optimal stack

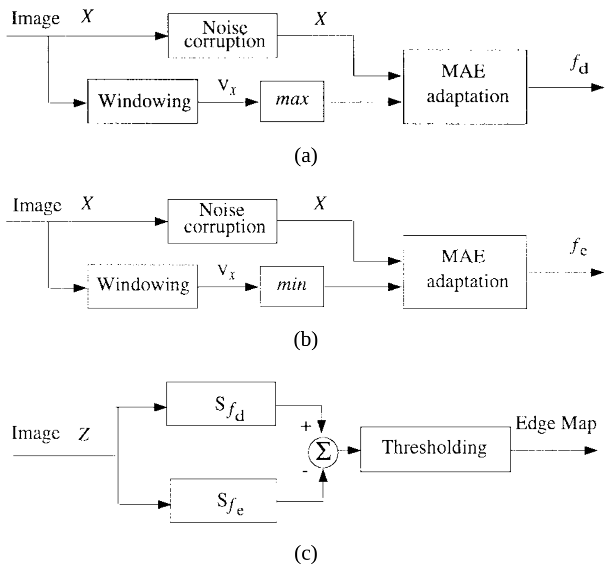


Fig. 3. Block diagram of the DoE edge detection scheme. (a) and (b) Data flow for training stack filters to serve the role of max (dilation) and min (erosion) operators, respectively. (c) Structure of the stack filter based DoE operator. Image Z may be an arbitrary observed image.

filter design. Therefore, we bound the total absolute error as follows:

$$\begin{aligned} E_{\mathrm{DoE}}(f_{\mathrm{d}}, f_{\mathrm{e}}) &= \sum_{s\in S} |\mathrm{DoE}_{\tilde{X}}(s) - D_X(s)| \\ &= \sum_{s\in S} |\{S_{f_{\mathrm{d}}}[W_{\tilde{X}}(s)] - S_{f_{\mathrm{e}}}[W_{\tilde{X}}(s)]\} \\ &\quad - [\max\{V_X(s)\} - \min\{V_X(s)\}]| \\ &\le \sum_{s\in S} |S_{f_{\mathrm{d}}}[W_{\tilde{X}}(s)] - \max\{V_X(s)\}| \\ &\quad + |S_{f_{\mathrm{e}}}[W_{\tilde{X}}(s)] - \min\{V_X(s)\}| \end{aligned} \tag{5}$$

where W is the window for the stack filters and V is the 2×2 square window for the max and min operators. The method of optimal stack filter design may now be used to minimize each of the two bounding terms in (5) by using two separate training operations. After the positive Boolean functions f_{d} and f_{e} are determined, the corresponding stack filters are used in the DoE edge operator as shown in Fig. 3.

To avoid having to train two separate stack filters for the DoE operator, and to further our understanding of the use of stack filters in edge detection, we propose a new method based on the duality of the optimal stack filters when the images used for training are symmetric. This new approach is called the Symmetric DoE (SDoE) approach. In the SDoE approach, we design a stack filter for the maximum (minimum) estimate with a symmetric image and then use its dual as the estimator for the minimum (maximum). The estimator for the minimum is then optimal because of the duality of the optimal stack filters proven in Theorem 2.1.

The advantages of the SDoE operator are that it can be designed with just one training run, and that it produces unbiased estimates because of the symmetry of the training data [7]. Now, the bound of the total absolute error given in (5) can be minimized by minimizing the total absolute error for either the maximum or the minimum.

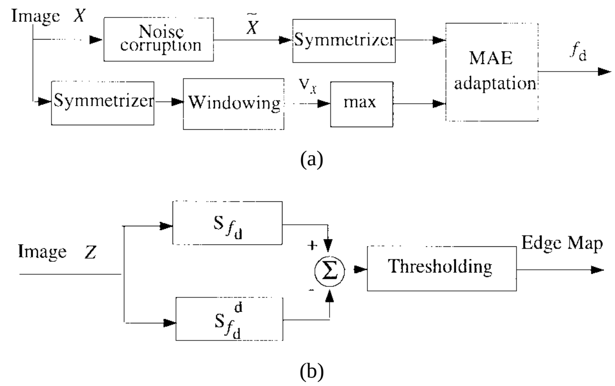


Fig. 4. Block diagram of the SDoE approach. (a) Data flow for training a stack filter to estimate the maximum and (b) Structure of the SDoE operator. The symmetrizer is used to generate symmetric training images.

If the procedure described in Section II-C is used to ensure that the images X and $\tilde{X}$ are symmetric, Corollary 2.1 can be used to bound the total absolute error of the SDoE operator:

$$\begin{aligned} E_{\mathrm{SDoE}}(f_{\mathrm{d}}) &\le 2 \sum_{s\in S} |S_{f_{\mathrm{d}}}[W_{\tilde{X}}(s)] - \max\{V_X(s)\}| \\ &= 2 \sum_{s\in S} |S_{f_{\mathrm{d}}^{\mathrm{d}}}[W_{\tilde{X}}(s)] - \min\{V_X(s)\}|. \end{aligned} \tag{6}$$

In (6), the positive Boolean function f_{d} for the optimal estimate of the maximum is designed and its dual $f_{\mathrm{d}}^{\mathrm{d}}$ is used to estimate the minimum. The corresponding stack filters $S_{f_{\mathrm{d}}}$ and $S_{f_{\mathrm{d}}^{\mathrm{d}}}$ are used for the optimal estimators in the DoE operator given in (4). This SDoE scheme for robust edge detection is shown in Fig. 4. The block labeled "symmetrizer" in this figure takes an input image X and puts out the union of X and its 256's complement, which is a symmetric image.

B. Dual Comparability

In this section, it is shown that the stack filters used in the SDoE approach are dual-comparable. As a result, the difference of the outputs of two stack filters used in the SDoE approach can be produced by a single operator that directly estimates the difference of the dilated and eroded versions of the noise-free original image. This single operator will be shown to obey the threshold decomposition property.

Some notation is needed in order to define dual comparability.

Definition 3.1: Let f and g be Boolean functions. If any binary vector a satisfying $f(a) = 1$ also satisfies $g(a) = 1$ but the reverse does not necessarily hold, we write

$$f \le g$$

and we say that g is implied by f [8].

Note that this extends the notation in (1) defining the stacking property of positive Boolean functions.

Definition 3.2: If there is an implication relation between a Boolean function f and its dual f^{d}, i.e., if either $f \le f^{\mathrm{d}}$ or $f \ge f^{\mathrm{d}}$ holds, then f is said to be dual-comparable and so is f^{d} [8].

Not all Boolean functions are dual-comparable. In fact, there are dual-comparable functions that are not unate and unate functions that are not dual-comparable [8]. Therefore, not all positive Boolean functions are dual-comparable, since they are a subclass of unate functions. For example, the following positive Boolean function f is not dual-comparable:

$$f(a_1, a_2, a_3, a_4) = a_1 \cdot a_2 + a_3 \cdot a_4$$

where $\cdot$ and + denote the logical *AND* and *OR*, respectively. To see this, note that $f(0, 0, 1, 1) = 1$ and $f^{\rm d}(0, 0, 1, 1) = 0$ while $f(0, 1, 1, 0) = 0$ and $f^{\rm d}(0, 1, 1, 0) = 1$.

The positive Boolean functions that arise in the SDoE approach are always dual comparable.

Theorem 3.1: Any positive Boolean function $f_{\rm d}(f_{\rm e})$ that is optimal for estimating the dilated (eroded) version of the original image in the SDoE approach is dual-comparable.

Proof: The outputs produced by the max and min filters when they are applied to the noise-free image are the desired images used in the training of $f_{\rm d}$ and $f_{\rm e}$, respectively. Note that since the same image is used as input for both the max and min filters, the output of the max filter is greater than that of the min filter at every point in the lattice T.

In the adaptive stack filtering algorithm [4], the element of the filter decision vector corresponding to the binary vector observed at the current threshold value and position of the filter window is incremented or decremented according to the pixel value at the same position in the desired image. If the pixel value in the desired image is greater than or equal to the threshold value, the element of the decision vector is incremented; otherwise, it is decremented. After a decision vector is updated, swap operations are performed to enforce the stacking constraint. Since each pixel value of the desired image for $f_{\rm d}$ is always greater than or equal to that for $f_{\rm e}$, each entry of the decision vector for $f_{\rm d}$ is always greater than the corresponding entry in the decision vector for $f_{\rm e}$. Then, since $f_{\rm d}^{\rm d}(\cdot) = f_{\rm e}(\cdot)$ by Theorem 2.1, it is always true that $f_{\rm d}(\cdot) \geq f_{\rm d}^{\rm d}(\cdot)$.

The dual of all the preceding statements shows that $f_{\rm e}$ is also dual-comparable. □

Theorem 3.1 can now be used to show that the output of the SDoE operator is the sum of the outputs produced when a nonpositive Boolean function is used on each threshold level of the operator. This implies that the SDoE operator obeys the threshold decomposition property even though it is not a stack filter.

Theorem 3.2: If $f_{\rm d}$ and $f_{\rm e}(= f_{\rm d}^{\rm d})$ are the positive Boolean functions defining the stack filters in the SDoE operator, then this operator can be represented by the sum of the outputs of the (possibly nonpositive) Boolean function $f_{\rm b} = f_{\rm d} - f_{\rm d}^{\rm d}$ as follows:

$$\begin{aligned}\text{SDoE}_{\tilde{X}}(s) &= S_{f_{\rm d}}[W_{\tilde{X}}(s)] - S_{f_{\rm d}^{\rm d}}[W_{\tilde{X}}(s)] \\ &= \sum_{l=1}^{M} f_{\rm b}[w_{\tilde{X}}(s, l)]\end{aligned}$$

where $\tilde{X}(s)$ is an observed image and $w_{\tilde{X}}(s, l)$ is the thresholded window vector of the image $\tilde{X}$. The SDoE operator thus satisfies the threshold decomposition property.

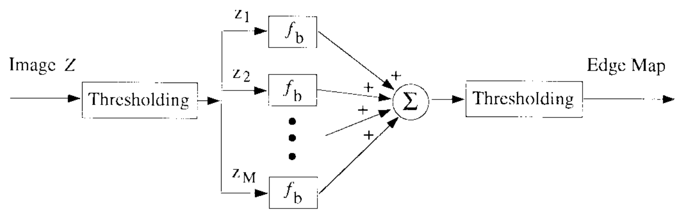


Fig. 5. Block diagram of the TBF operator based on the Boolean function $f_{\rm b}$. Image Z is an arbitrary observed image.

Proof: Since $f_{\rm d}$ is dual-comparable and is implied by its dual $f_{\rm d}^{\rm d}$, $f_{\rm d}(\cdot) \geq f_{\rm d}^{\rm d}(\cdot)$ is always true. Therefore,

$$\begin{aligned}&S_{f_{\rm d}}[W_{\tilde{X}}(s)] - S_{f_{\rm d}^{\rm d}}[W_{\tilde{X}}(s)] \\ &\quad= \sum_{l=1}^{M} f_{\rm d}[w_{\tilde{X}}(s, l)] - f_{\rm d}^{\rm d}[w_{\tilde{X}}(s, l)] \\ &\quad= \sum_{l=1}^{M} |f_{\rm d}[w_{\tilde{X}}(s, l)] - f_{\rm d}^{\rm d}[w_{\tilde{X}}(s, l)]| \\ &\quad= \sum_{l=1}^{M} f_{\rm d}[w_{\tilde{X}}(s, l)] \oplus f_{\rm d}^{\rm d}[w_{\tilde{X}}(s, l)] \\ &\quad= \sum_{l=1}^{M} f_{\rm b}[w_{\tilde{X}}(s, l)].\end{aligned}$$

Since $|f - g| = f \oplus g$ for any two Boolean functions f and g, it is always possible to find a Boolean function $f_{\rm b}$ which is equivalent to the *exclusive-OR* of $f_{\rm d}$ and $f_{\rm d}^{\rm d}$. □

Since the Boolean function $f_{\rm b}$ in Theorem 3.2 is not necessarily positive, the corresponding multilevel operator is not necessarily a stack filter. As shown in Fig. 5, the output of the SDoE operator is obtained by decomposing an input image into a set of binary images, carrying out the binary filtering operation with the Boolean function $f_{\rm b}$ on each threshold level, and then summing up the results. The SDoE operator thus satisfies the threshold decomposition property, which means that it is a TBF as defined in [2]. This corresponding multilevel operation is, of course, the difference between the outputs of the stack filters defined by $f_{\rm d}$ and $f_{\rm e}$.

C. Threshold Boolean Function Approach

Theorem 3.2 at the end of the preceding subsection suggests that an edge operator could be designed directly by assuming it has the threshold decomposition architecture and then finding a (possibly nonpositive) Boolean function to put on each level of this architecture. The resulting operator is known as a Threshold Boolean Function (TBF) [2], and when this operator is designed to estimate gradients, we call it the TBF approach to edge detection.

Since the TBF approach subsumes the SDoE approach, it should be able to produce operators that yield a better estimate of the gradient. Unfortunately, designing a TBF to minimize the mean absolute error or some other standard error criterion is difficult, as discussed in [3]. The problem encountered with the mean absolute error criterion is that the loss of the stacking constraint means the multilevel absolute error can not be

(a)

(b)

Fig. 6. Training images used in experiments. (a) Noise corrupted training image $couple_{I10}$. The noise is impulsive with an occurrence probability of 0.1 and an absolute magnitude of 200. (b) Inverse of (a).

decomposed into the sum over all levels of the absolute error on each level of the threshold decomposition architecture.

If, however, a nonstandard error criterion is used, the optimization problem becomes tractable. The new error criterion is the sum, over all levels, of the absolute error on each threshold level. Note that the multilevel mean absolute error is always upper bounded by the sum of the mean absolute error on each threshold level. This new strategy could then be viewed as minimizing a bound of the absolute error

$$\begin{aligned} E_{\text{TBF}}(f) &= \sum_{s\in S} |\text{TBF}_{\tilde{X}}(s) - D_X(s)| \\ &= \sum_{s\in S} \Bigg| \sum_{l=1}^{M} \{f[w_{\tilde{X}}(s, l)] \\ &\quad -[\max\{v_X(s, l)\} - \min\{v_X(s, l)\}]\} \Bigg| \\ &\le \sum_{s\in S} \sum_{l=1}^{M} | f[w_{\tilde{X}}(s, l)] \\ &\quad - [\max\{v_X(s, l)\} \oplus \min\{v_X(s, l)\}]|. \quad (7) \end{aligned}$$

The problem of finding a TBF which is best under this error criterion can be solved with a simple modification of the minimum mean absolute error (MMAE) filter design techniques already developed for stack filters [4]. The stacking constraint is no longer required, so the swap operations in the algorithm can be eliminated. The resulting algorithm will clearly run faster than the algorithm for training a stack filter. As will be seen in the next section, though, the TBF approach does not always lead to lower absolute error than the DoE approach, and even when it does, the loss of the stacking constraint leads to filters that are not robust.

IV. Experimental Results

It has already been shown that the DoE approach works very well when compared to a variety of existing methods for edge detection [1]. Therefore, in the experiments described here, we just compare the performance of the TBF operators to the performance of the DoE and SDoE operators. The robustness of each method is also tested by training filters in each approach with images and noise distributions that are different from those used for testing.

A. Designing the Edge Operators

The training of the *two* stack filters in the DoE scheme was done with the 256 $\times$ 256 natural image "couple" shown in Fig. 2(a) and the noisy version shown in Fig. 6(a), which was corrupted by impulsive noise with occurrence probability of 0.1 and an absolute magnitude of 200. We denote this noise corrupted image $couple_{I0.1}$. A *pair* of 4 $\times$ 4 stack filters were designed to estimate the results of the max or min operations with 2 $\times$ 2 square windows on the original noiseless image. The resulting operator is called *DoE.*

A Boolean function of 16 variables (a 4 $\times$ 4 square window), which characterizes a multilevel threshold Boolean function in the TBF approach was also designed with the training data, $couple_{I0.1}$, used to train the DoE operator. The operator that was produced is called TBF_{asym}. The adaptive algorithm from [4]—without the swap operations—was used to design this operator. The Boolean function of 16 variables was directly designed to estimate the difference of the outputs of the max and min operators with 2 $\times$ 2 square windows on the original noiseless image. As mentioned in Section III-C, the computational requirements of the adaptive algorithm used in the TBF approach is significantly lower than that of the adaptive stack filtering algorithm used in the DoE scheme because the swap operations are not performed.

In the SDoE approach, one stack filter must be trained with the algorithm in [4]. The images shown in Figs. 2 and 6 were used to train a 4 $\times$ 4 stack filter to estimate the result of the max operation with a 2 $\times$ 2 square window on the original noiseless image. Its dual was then used for the optimal estimate

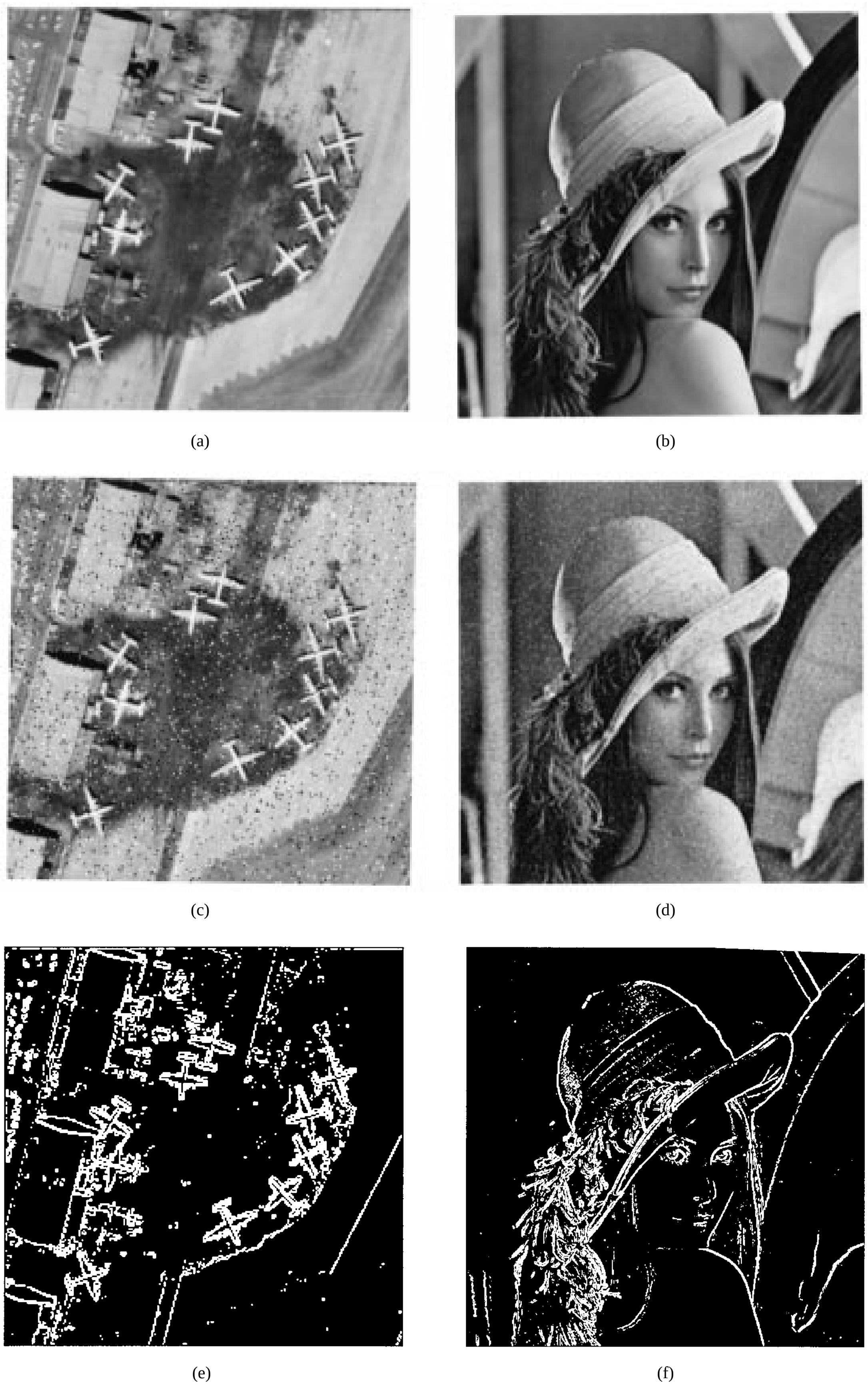
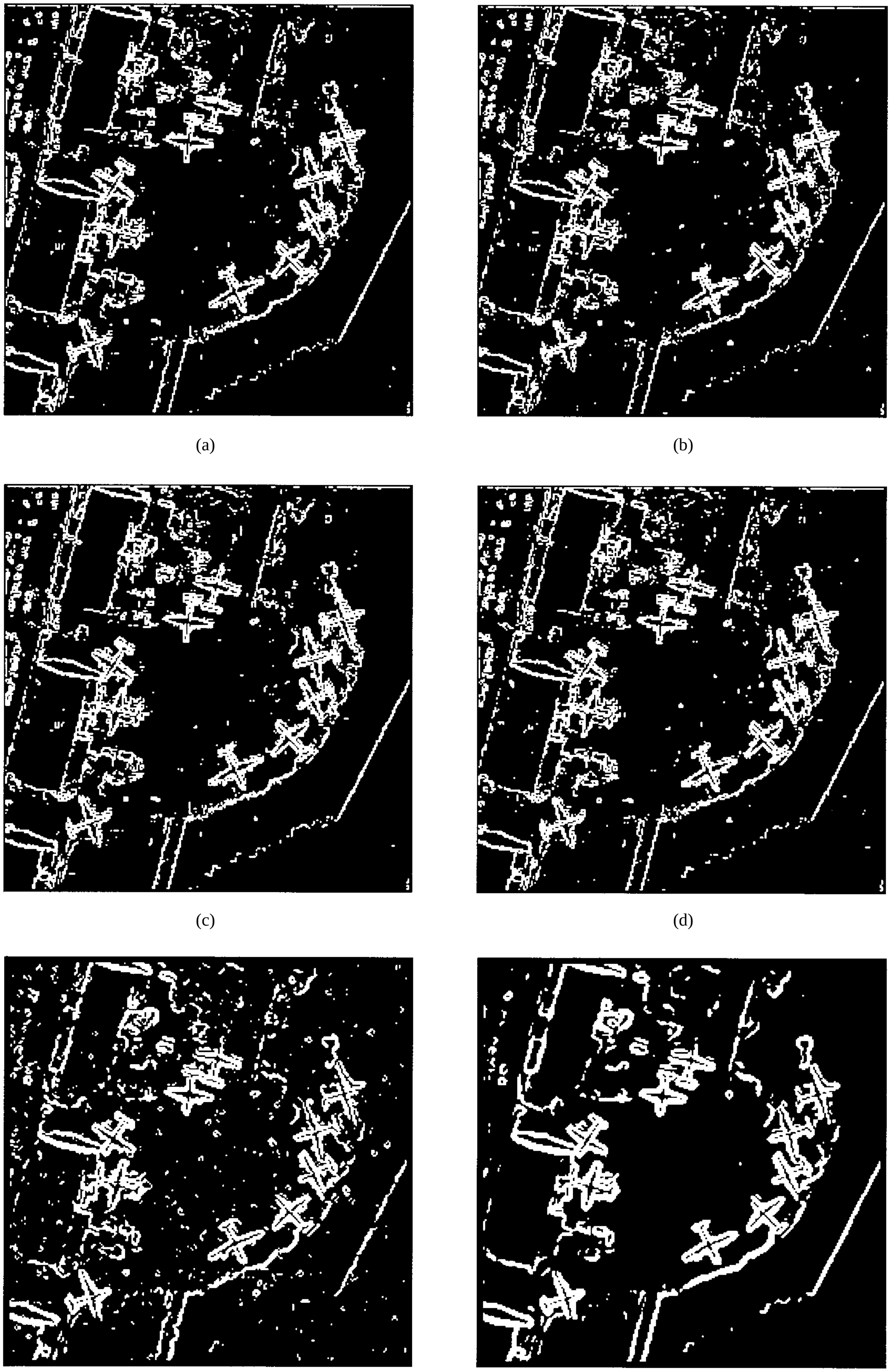
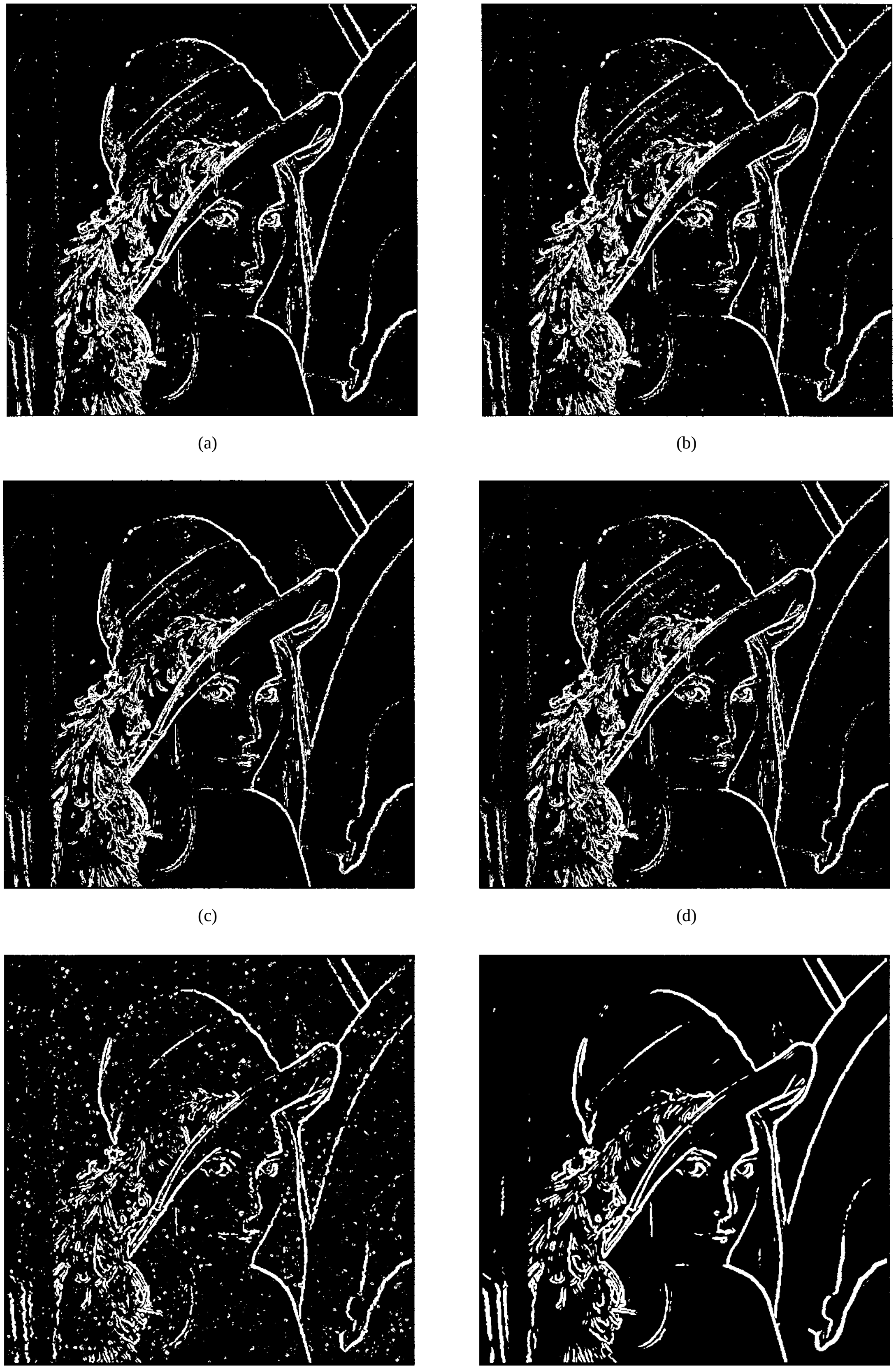


Fig. 7. Test images used in experiments. (a) Aerial photograph with 256 × 256 resolution. (b) Lenna with 512 × 512 resolution. (c) $Aerial_{I0.05G7.9}$. (d) $Lenna_{I0.05G7.9}$. (e) Reference edge map of (a). (f) Reference edge map of (b). Test images in (c) and (d) are simultaneously corrupted by impulse noise with an occurrence probability of 0.05, and by Gaussian noise with a standard deviation of 7.9.



(a) (b) (c) (d) (e) (f)

Fig. 8. For the test image $aerial_{I0.05G7.9}$, results of applying the (a) DoE operator, (b) TBF_{asym} operator, (c) SDoE operator, (d) TBF_{sym} operator, (e) Canny operator, and (f) prefiltered Canny operator with 3×3 median filter.



(a) (b) (c) (d) (e) (f)

Fig. 9. For the test image $Lenna_{I0.05G7.9}$, results of applying the (a) DoE operator, (b) TBF_{asym} operator, (c) SDoE operator, (d) TBF_{sym} operator, (e) Canny operator, and (f) prefiltered Canny operator with 3 × 3 median filter.

TABLE I
COMPARISON OF THE MEAN ABSOLUTE ERRORS FOR THE DoE AND TBF SCHEMES WITH BOTH SYMMETRIC AND ASYMMETRIC TRAINING IMAGES

Training image	Operator	Absolute Error
Asymmetric	DoE	2.89
	TBF_{asym}	2.65
Symmetric	$SDoE$	2.93
	TBF_{sym}	2.80

TABLE II
COMPARISON OF ERRORS INCURRED BY THE DoE AND TBF SCHEMES FOR VARIOUS TRAINING AND TEST IMAGES

	$Aerail_{I0.05G7.9}$			$Lenna_{I0.05G7.9}$		
Training Image	DoE	$SDoE$	TBF	DoE	$SDoE$	TBF
Asymmetric $Couple$	7.25	-	7.34	4.93	-	5.23
Symmetric $Couple$	-	7.30	7.33	-	4.94	5.10
Test Image	7.05	-	6.39	4.64	-	4.69

of the output of the min operator. The resulting operator is called *SDoE*.

The advantage of the SDoE operator over the DoE operator is that only one filter must be trained in the SDoE approach, while two must be trained in the DoE approach. Therefore, the computational costs of designing an SDoE operator are as little as half of that of a DoE operator when the same size training image is used to design both operators.

Finally, a TBF operator with a 4 × 4 square window was designed with the same symmetric training data used to train the SDoE operator. The operator which was produced is called TBF_{sym}.

B. Performance Comparison

Table I shows the minimum average absolute error per pixel in the estimates of the maximum of the gradient produced by the edge operators designed in the previous subsection when these operators are applied to the *training* image $couple_{I0.1}$.

As shown in Table I, TBF operators yielded less error than those of the DoE and SDoE operators for this specific set of training images. This is not always the case. As shown in Table II, the DoE operator outperforms the TBF operator when a different training image is used. The reason for this behavior is that both operators only minimize an upper bound on the error for estimating the maximum gradient in the window. In the case of the TBF, the upper bound is that given in (7); in the case of the DoE, the upper bound is that given in (5). The speed with which the TBF filter can be designed, though, would seem to outweigh this consideration.

It is important to note that the edge operators were compared against each other by applying them to the images used to train them. This gives some idea of the relative performance of the operators when the type of image and noise encountered are the same as those used in the design process. It does not, however, test these operators for robustness. Such tests are critical since the target image and the noise corrupting it may, for a variety reasons, differ significantly from those used in training. In this section, such tests are also carried out.

Two natural images, the aerial image and the Lenna photographs shown in Fig. 7(a) and (b), were used to test the DoE, SDoE, and TBF operators. As a benchmark, the Canny operator [9] was also tested—elaborate comparisons between the DoE and other edge operators can be found in [1].

The noisy images in Fig. 7(c) and (d) were formed by adding both impulsive noise with an occurrence probability of 0.05 and absolute magnitude of 200, and Gaussian noise with standard deviation of 7.9 to the noiseless original images in Fig. 7(a) and (b). These two noisy images are called $aerial_{I0.05G7.9}$ and $Lenna_{I0.05G7.9}$, respectively.

The reference edge maps (from the noise-free images) for the aerial photograph and Lenna are shown in Fig. 7(e) and (f). They were produced by setting threshold values so that the largest 12% and 8% gray-value pixels were selected for the aerial photograph and Lenna, respectively.

Fig. 8(a)–(f) are the results of the DoE, TBF_{asym}, SDoE, TBF_{sym}, Canny, and prefiltered Canny approaches with the test image $aerial_{I0.05G7.9}$. The threshold for edge detection was set at a level which selected the largest 12% of gray-value pixels for each operator. A 3 × 3 median filter was used as a prefilter with the Canny operator. Fig. 9(a)–(f) shows the results when the same operators are applied to the test image $Lenna_{I0.05G7.9}$ with an 8% threshold.

The difference in the noise sensitivity of the DoE and SDoE operators is difficult to detect in the test images. Both operators exhibit a significant degree of robustness; they perform about as well on the test images as on the training image. The TBF operator, on the other hand, shows a lack of robustness. The edge maps it produces when the target image is different from the training image are noisier than those obtained from either the DoE or SDoE operators. Note, though, that despite its lack of robustness when compared with the DoE operator, the TBF operator produces much better edge maps than the Canny operator when impulsive noise is present. Even though the prefiltered Canny operator with 3 × 3 median filter has less noise sensitivity, it loses the most detail in the reference edge maps of the test images shown in Fig. 7(e) and (f). In particular, notice that the engines and the shadows of the wings in Fig. 7(e) have been more clearly detected by the DoE and TBF operators than by the Canny and prefiltered Canny operators.

As a final result, Table II shows the error incurred by the DoE and modified DoE operators for the different images shown above, and for the case in which the filters were trained on these images instead of on the couple image. The last two entries in the bottom row show that the TBF operator is not always better than the DoE operator when they are applied to the image used to train them. The robustness of the DoE operator is shown in the errors given in the first two rows, which correspond to operators trained on the image couple. The error for the DoE and SDoE operators is significantly less than that of the TBF operator.

The results of these experiments indicate that if sufficient time is available to train a DoE or SDoE operator, then they should be used instead of the TBF operator. Recent work on the stack filter training algorithm has resulted in significant speedups, particularly if a parallel machine such as a MASPAR MP-1 is available. In fact, DoE operators with 4 × 4 windows can now be trained in less than a minute. If sufficient time for training the SDoE operator is still not available, then the TBF operator could be used. One would then have to hope that the target image and noise are not much different than those used in the training stage.

There is thus a trade-off to be faced when choosing between the TBF approach and the DoE approaches. If it is known that the training image is very similar to the target image, and that the noise used in training is similar to that expected in the target image, then the TBF approach could be used since it offers reduced training time. If the target image may differ significantly from the training image, or if some mismatch in noise is possible between the training noise and the noise that might be encountered, one of the DoE approaches should be used.

V. Conclusion

Perhaps the most important result in this paper is the demonstration—on real images—that the DoE and SDoE operators exhibit robustness. This implies that the stacking property ensures robustness. This is satisfying from a theoretical point of view since it confirms that combinations and cascades of rank order operators exhibit the same robustness as single rank operators.

References

[1] J. Yoo, C. A. Bouman, E. J. Delp, and E. J. Coyle, "The nonlinear prefiltering and difference of estimates approaches to edge detection: Application of stack filters," *CVGIP: Graph. Models Image Process.*, vol. 55, Mar. 1993.

[2] J. Song and Y. H. Lee, "Linear combination of weighted order statistic filters: Canonical structure and optimal design," *IEEE Trans. Circuits Syst.*, vol. 43, May 1996.

[3] K. D. Lee and Y. H. Lee, "Optimization of threshold Boolean filters," in *IEEE Winter Workshop on Nonlinear Digital Signal Processing*, Tampere, Finland, Jan. 1993.

[4] 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.

[5] 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.

[6] E. N. Gilbert, "Lattice-theoretic properties of frontal switching functions," *J. Math Phys.*, vol. 33, pp. 57–67, Apr. 1954.

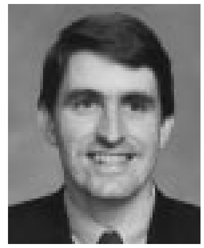
[7] O. Yli-Harja, J. Astola, and Y. Neuvo, "Analysis of the properties of median and weighted median filters using threshold logic and stack filter representation," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 39, pp. 395–410, Feb. 1991.

[8] S. Muroga, *Threshold Logic and Its Application.* New York: Wiley, 1971.

[9] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-8, pp. 679–698, Nov. 1986.

Jisang Yoo was born in Seoul, Korea, in 1962. He received the B.S. and M.S. degrees from Seoul National University, Seoul, Korea, and the Ph.D. degree from Purdue University, West Lafayette, IN, all in electronics engineering, in 1985, 1987, and 1993, respectively.

From September 1993 to August 1994, he worked as a Senior Research Engineer in the Industrial Electronics R&D Center, Hyundai Electronics Industries Co., Ltd., Ichon, Korea, in the area of image compression and HDTV. From 1994 to 1997, he was an Assistant Professor with the Department of Electronics Engineering, Hallym University, Chuncheon, Korea. He is currently an Assistant Professor with the Department of Electronics Engineering, Kwangwoon University, Seoul. His research interests are in signal and image processing and computer graphics.



Edward J. Coyle (S'79–M'82–SM'93) received the B.S.E.E. degree from the University of Delaware, Newark, in 1978, and the Ph.D. degree in electrical engineering and computer science from Princeton University, Princeton, NJ, in 1982.

Since 1982, he has been with the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN, where he is currently a Professor and Co-director of the Engineering Projects in Community Service (EPICS) Center. His research interests are in digital signal and image processing, computer networks, and engineering education.

Dr. Coyle has served as an Associate Editor for the IEEE Transactions on Circuits and Systems and as a member of the Board of Governors of the IEEE Circuits and Systems Society. He was a co-recipient of the 1986 ASSP Best Paper Award for authors under the age of 30, and was a co-recipient of the Myril B. Reed Best Paper Award from the 32nd Midwest Symposium on Circuits and Systems. For his work as a co-founder and co-director of the EPICS Program in the Schools of Engineering, Purdue University, he was co-recipient of the 1997 Chester F. Carlson Award for Innovation in Engineering Education of the American Society for Engineering Education.

Charles A. Bouman (S'86–M'89) received the B.S.E.E. degree from the University of Pennsylvania, Philadelphia, in 1981, and the M.S. degree in electrical engineering from the University of California at Berkeley in 1982. In 1987 and 1989, respectively, he received the M.A. and Ph.D. degrees in electrical engineering from Princeton University, Princeton, NJ, under the support of an IBM graduate fellowship.

From 1982 to 1985, he was a Staff Member in the Analog Device Technology Group, Massachusetts Institute of Technology Lincoln Laboratory, Lexington, MA. In 1989, he joined the faculty of Purdue University, West Lafayette, IN, where he currently holds the position of Associate Professor in the School of Electrical and Computer Engineering. His research interests include statistical image modeling and analysis, multiscale processing, and the display and printing of images. He is particularly interested in the applications of statistical signal processing techniques to problems such as fast image search and browsing, tomographic reconstruction, document segmentation, and image rendering. He has performed research for numerous government and industrial organizations including National Science Foundation, the U.S. Army, Hewlett-Packard, NEC Corporation, Apple Computer, Xerox, and Eastman Kodak.

Dr. Bouman is a Member of SPIE and IS&T professional societies. From 1991 through 1993, he was also an NEC Faculty Fellow. He has been both chapter chair and vice chair of the IEEE Central Indiana Signal Processing Chapter, and an associate editor of the IEEE Transactions on Image Processing. He is currently a member of the IEEE Image and Multidimensional Signal Processing Technical Committee.