Proc. Natl. Sci. Counc. ROC(A)
Vol. 20, No. 5, 1996. pp. 550-563

Finding of Optimal Morphological Erosion Filter by Greedy and Constraint Searching Algorithms

CHIN-CHUAN HAN* AND KUO-CHIN FAN**,†

**Institute of Information Science*
Academia Sinica
Taipei, Taiwan, R.O.C.
***Institute of Computer Science and Information Engineering*
National Central University
Chung-Li, Taiwan, R.O.C.

(Received November 30, 1995; Accepted May 27, 1996)

ABSTRACT

A graphic-theoretic search-based algorithm is proposed to find the optimal binary morphological erosion filter. According to the Matheron representation, the binary morphological erosion filter is defined as the union of multiple erosions. Traditionally, finding the optimal solution involves a long search and is a time consuming procedure because we have to compute the MSE values over all possible structuring element combinations and make comparisons among them. The search for the optimal solution is reduced to the problem of obtaining a path with the minimal cost from the root node to the optimal node on an error code graph (ECG). In this paper, two graphical searching techniques, the *greedy* and *constraint satisfaction algorithms*, are applied to avoid searching an extremely large search space. Experimental results are illustrated to show the feasibility and efficiency of our proposed method.

Key Words: morphological erosion filter, error code graph, mean-square error, greedy searching, constraint satisfaction searching

I. Introduction

Optimal mean-square morphological filters, both binary and gray scale, have been characterized by Dougherty (1992) and Dougherty *et al.* (1991) in terms of the Matheron representation. In the binary morphological erosion case, Dougherty (1992) applied erosion as an estimator to find the optimal filter automatically by minimizing the mean-square error (MSE). Loce and Dougherty (1992a) also proposed the erosion operation as the estimation rule to minimize the mean-absolute error (MAE). Both of these estimation rules have the same effects on binary images. However, the estimation process, even in the binary case, is extremely time consuming because finding the optimal mean-square or mean-absolute morphological filter requires searching a large number of structuring element combinations. As mentioned by Dougherty (1992), the computation time grows rapidly with the size of the observation window, and it is almost impossible to design a filter without using a high-performance computer.

Various approaches to decreasing the searching time in the optimal binary morphological erosion filter have been proposed. An approach to increasing the computational efficiency was given by Loce and Dougherty (1991, 1992b) which involves constraining the class of filters from which the optimal morphological filter is to be chosen. The constraint techniques involve limiting the structuring element's size and shape, the number of structuring elements forming the filter, and the search of the structuring element from some previously chosen library. In considering these perspectives, Dougherty's proposed approach only provides a sub-optimal filter. Dougherty, Mathew and Swarnaker (Dougherty *et al.*, 1991; Mathew *et al.*, 1993; Dougherty and Loce, 1993) developed an algorithmic approach to performing the search of optimal filters. They derived the optimal morphological filter from the conditional expectation, from which the *switching* methodology offers quite good computational efficiency for certain types of noise conditions.

A graphic search-based method was proposed by

†To whom all correspondence should be addressed.

Han and Fan (1994) to decrease the searching time for the optimal solution. They designed the error code graph (ECG) to find the optimal morphological erosion filter by making use of the *greedy* and *branch and bound* (B&B) techniques. The problem of finding the optimal solution is reduced to the problem of searching for a shortest path in the ECG. Since ECG satisfies some greedy properties, only a few nodes need to be traversed and examined in the graph. Furthermore, three heuristic methods have also been proposed to mitigate the worst case of the B&B method.[1] Though the obtained solution via the three heuristic methods can not guarantee that a global optimal filter will be found, the obtained local optimal filter is a global optimal filter in many image cases.

In this paper, the graphic search-based algorithm proposed by Han and Fan (1994) is further improved by utilizing the *greedy* and *constraint satisfaction* searching algorithms. In the artificial intelligence (A.I.) field (Rich, 1983), many problems are considered as problems of constraint satisfaction in which the goal is to find some problem states that satisfy a given set of constraints. Three inference rules are designed to choose the correct candidate vector for the undecidable case of the greedy property. The vector satisfying the inference rules is chosen as an element of the optimal path in the ECG. We also guarantee that the found filter is a global optimal solution. Furthermore, there are at most 2^N-1 nodes to be traversed in the ECG.

The rest of this paper is organized as follows. Section II gives a brief discussion of the fundamental propositions of the morphological filter via minimization of the mean-square error. The error code graph and its *greedy properties* are also recalled in this section. The improvement of the searching algorithm is described in Section III. In this section, three inference rules are developed which guarantee that the solution we find is actually an optimal solution. Section IV gives some experimental results to show the efficiency of our proposed algorithm. Moreover, the time complexity and memory space requirement of the proposed method are analyzed in Section IV. Finally, conclusions are given in Section V.

II. Background

In this section, the problem of the optimal binary morphological erosion filter is firstly presented, and then the searching process for the optimal solution carried out by comparing all the MSE values of possible basis element combinations is described. To improve the performance of the searching process, a graphic search-based method via the greedy and B&B techniques proposed by Han and Fan (1994) is briefly described in subsections II.2 and II.3.

1. Problem Statements

Dougherty (1992) designed an optimal binary morphological erosion filter in order to obtain the best outcome of erosion operation. He used the erosion operation as an estimator to design the optimal filter which minimizes the mean square error on binary images. According to the Matheron representation, the estimator for a general filter is defined as the union of multiple erosions. In Dougherty (1992) and Loce and Dougherty (1992a), the general filter containing M structuring elements is expressed as follows:

$$\Psi(\tilde{x}) = \bigcup_i \{\tilde{x} \ominus A(i)\} = \max_{i=1}^{M}\left\{\min_{j=1}^{N}\{x[j] : a[i,j]=1\}\right\}, \quad (1)$$

where $\tilde{x}=(x[1], x[2], ..., x[N])$ is an N-tuple binary vector in the universal set $\mathbf{U}$ that is generated from N random variables $X[1], X[2], ..., X[N]$. The collection $\{A(i)\}=\{a[i,j]: i=1...M, j=1...N\}$ is called the *basis* for Ψ, denoted as $\text{Bas}(\Psi)$, which satisfies the *morphological basis criteria*: **No element of** $\text{Bas}(\Psi)$ **is a proper sub-image of any other element of** $\text{Bas}(\Psi)$ (Dougherty and Haralick, 1991b; Maragos, 1989).

In two-dimensional image noise filtering, N random variables are considered as the functions of a window mask with size N which slides the image from left to right and top to bottom. For example, a 3×3 window mask is a filter of size 9, where the origin is located at the center of window mask. Similarly, the basis of a morphological filter is the collection of window masks which are operated on the image.

The value of MSE for a general N-observation morphological erosion filter in Dougherty (1992) is defined as

$$\begin{aligned}
&\text{MSE}\langle\Psi\rangle \\
&= E\left[|z - \psi(X[1], X[2], \ldots, X[N])|^2\right]; \\
&= E\left[\left|Z - \max_{i=1}^{M}\left\{\min_{j=1}^{N}\{x[j] : a[i,j]=1\}\right\}\right|^2\right]; \\
&= \sum\left\{f(x[1], x[2], \ldots, z) : z \neq \max_{i=1}^{M}\left\{\min_{j=1}^{N}\{x[j] : a[i,j]=1\}\right\}\right\}, \quad (2)
\end{aligned}$$

[1] Han, C. C. and K. C. Fan (1996) Finding of optimal binary morphological erosion filter by heuristic search methods. Submitted to *Cricuit, Systems and Signal Processing*.

where $f(x[1], x[2], ..., x[N], z)$ is the density value for the N-observation random variables and the estimated random variable Z. Finding the optimal N-observation filter is now reduced to the problem of selecting the subset of 2^N structuring elements that yields minimum MSE<Ψ>. We need a tremendous amount of time to compute the MSE values over all the possible basis combinations and to make comparisons among them to pick out the optimal solution. To illustrate the generation of an optimal binary morphological erosion filter, let us consider a three-observation filter in the following example.

Example 1. The probabilities and density values $f(x[1], x[2], x[3], z)$ which are generated by three random variables $X[1]$, $X[2]$, $X[3]$ and an estimated random variable Z are given in Table 1(a). From Eq. (2), errors will occur in the situation where the estimated realization z differs from the eroded result indicated by symbol 'x' in Table 1(b) and where the MSE value for each basis element can be calculated by summing the density values marked by 'x'. Thus, the MSE values for all possible basis element combinations are computed and listed in the bottom row of Table 1(b). In this illustration, the basis element {100} with the smallest MSE value 0.636 is assigned as the optimal morphological erosion filter.

2. Construction of Error Code Graph

A graphic search-based method was proposed by Han and Fan (1994) to decrease the searching time for an optimal morphological erosion filter. Now, we will give a brief review of the graphic search-based algorithm in this subsection.

From Eq. (1), the errors eroded by a basis $\mathrm{B{\scriptsize AS}}(\Psi)$ indicated by symbol 'x' in Table 1(b) for each binary input realization are encoded as the attribute $E\hat{C}V$ of

Table 1. (a) The Probabilities and Density Values (b) The Errors of Erosion for a 3-observation Filter

(a)

$x[1], x[2], x[3], z$	Probabilities	$x[1], x[2], x[3], z$	Probabilities
0000	0.156	1000	0.061
0001	0.301	1001	0.547
0010	0.020	1010	0.008
0011	0.034	1011	0.061
0100	0.480	1100	0.188
0101	0.018	1101	0.033
0110	0.062	1110	0.024
0111	0.002	1111	0.004

(b)

	Basis																		
$\tilde{x}$, z	000	001	010	011	100	101	110	111	001 010	001 100	001 110	010 100	010 101	011 100	011 101	011 110	101 110	011 101 110	001 010 100
0000	x																		
0001		x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0010	x	x							x	x	x								x
0011			x	x	x	x	x	x				x	x	x	x	x	x	x	
0100	x		x						x			x	x						x
0101		x		x	x	x	x	x		x	x			x	x	x	x	x	
0110	x	x	x	x					x	x	x	x	x	x	x	x		x	x
0111					x	x	x	x									x		
1000	x				x					x		x		x					x
1001		x	x	x		x	x	x	x		x		x		x	x	x	x	
1010	x	x			x	x			x	x	x	x	x	x	x		x	x	x
1011			x	x			x	x								x			
1100	x		x		x		x		x	x	x	x	x	x		x	x	x	x
1101		x		x		x		x							x				
1110	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
1111																			
MSE	–	1.013	1.697	1.080	0.636	0.967	1.175	1.020	1.630	0.682	1.168	1.158	1.644	0.696	1.027	1.235	1.123	1.182	1.144

$\tilde{x} = (x[1], x[2], x[3])$

Table 2. The Error Code Value $E\hat{C}V$ for a 3-observation Morphological Filter

realization	$E\hat{C}V$									
basis	111	110	101	100	011	010	001	000	$E\hat{C}V_{(16)}$	MSE
001	0	1	0	1	0	1	0	1	55	1.013
010	0	0	1	1	0	0	1	1	33	1.697
011	0	1	1	1	0	1	1	1	77	1.080
100	0	0	0	0	1	1	1	1	0F	0.636
101	0	1	0	1	1	1	1	1	5F	0.967
110	0	0	1	1	1	1	1	1	3F	1.175
111	0	1	1	1	1	1	1	1	7F	1.020
001 010	0	0	0	1	0	0	0	1	11	1.630
001 100	0	0	0	0	0	1	0	1	05	0.682
001 110	0	0	0	1	0	1	0	1	15	1.168
010 100	0	0	0	0	0	0	1	1	03	1.158
010 101	0	0	0	1	0	0	1	1	13	1.644
011 100	0	0	0	0	0	1	1	1	07	0.696
011 101	0	1	0	1	0	1	1	1	57	1.027
011 110	0	0	1	1	0	1	1	1	37	1.235
101 110	0	0	0	1	1	1	1	1	1F	1.123
011 101 110	0	0	0	1	0	1	1	1	17	1.182
001 010 100	0	0	0	0	0	0	0	1	01	1.144
$f(\bar{x},0)$	0.024	0.188	0.008	0.061	0.062	0.480	0.020	0.156		
$f(\bar{x},1)$	0.004	0.033	0.061	0.547	0.002	0.018	0.034	0.301		
$-f(\bar{x},1)+f(\bar{x},0)$	0.020	0.155	-0.053	-0.486	0.060	0.462	-0.014	-0.145		

BAS(Ψ) . These $E\hat{C}V$ and MSE values are also listed in the rightmost two columns of Table 2. Second, an *error code graph* (ECG) is constructed as a directed graph with 2^N-1 vertical levels for an N-observation filter. In the ECG, the basis elements are written inside the circle for each vertex and the attribute $E\hat{C}V$ expressed in hexadecimal form is shown below the circle. Only a vertex called the maximum node with an error code vector of one '0' and 2^N-1 '1' bits is located in the leftmost level, called *level* 2^N-1 whereas a minimum node with 2^N-1 '0' and a '1' is located in the rightmost level, called *level* 1. Furthermore, only one bit is converted between the attributes of two contiguous levels.

Traditionally, it seems that we should construct the ECG first and then find the optimal solution from this constructed graph. Actually, it is not necessary to construct the ECG before searching for the optimal solution. Consider the current basis element $\mathrm{BAS}_a(\Psi)$ and its successor $\mathrm{BAS}_b(\Psi)$; the basis element of node $\mathrm{BAS}_b(\Psi)$ can be obtained based on the following manipulation:

$$\mathrm{BAS}_b(\Psi) = \mathrm{RED}(\mathrm{EXT}(\mathrm{BAS}_a(\Psi)) \cup \{\bar{x}\}),$$

$$\text{and } \bar{x} \in \mathrm{CAN}(\mathrm{BAS}_a(\Psi)), \tag{3}$$

where

$\mathrm{EXT}(\mathrm{BAS}(\Psi)) = \{\bar{x} : \bar{x} \wedge \bar{y} = \bar{y}, \bar{y} \in \mathrm{BAS}(\Psi), \bar{x} \in U\}$,
$\mathrm{RED}(A) = A \setminus \{\bar{x} : \bar{x} \in A, \exists \bar{y} \in A, \bar{x} \neq \bar{y}, \bar{x} \wedge \bar{y} = \bar{y}\}$,
and
$\mathrm{CAN}(\mathrm{BAS}_a(\Psi)) = \{\bar{x} : \bar{x} \in U \setminus \mathrm{EXT}(\mathrm{BAS}_a(\Psi)), \mathrm{EXT}(\{\bar{x}\}) \setminus \{\bar{x}\} \subseteq \mathrm{EXT}(\mathrm{BAS}_a(\Psi)), \bar{x} \neq \text{null vector}\}$.

If we want to find the optimal basis whose MSE value is a minimum, we can search the ECG to find an *optimal path* with the minimal cost from the root vertex $\mathrm{BAS}_{max}(\Psi)$. Thus, the MSE value for the *optimal node* is

$$\mathrm{MSE}\langle \mathrm{BAS}_{optimal}(\Psi)\rangle$$

$$= \mathrm{MSE}\langle \mathrm{BAS}_{max}(\Psi)\rangle + \text{the cost of path from node}$$

$$\mathrm{BAS}_{max}(\Psi) \text{ to } \mathrm{BAS}_{optimal}(\Psi). \tag{4}$$

3. Graph Searching Algorithm via the Greedy and Branch and Bound Techniques

Traditionally, a search for an optimal solution in graph theory needs to traverse the arcs and vertices at least once. Fortunately, since the ECG satisfies some greedy properties which will be described below, we do not need to traverse all the vertices and arcs.

Greedy Property: While searching the ECG, if the current vertex $\mathrm{BAS}_a(\Psi)$ is on the optimal path and at least one vertex is connected to the current vertex via a negative cost arc, which is called the *greedy condition*, then the successor traversed via the arc with the smallest negative cost is also on the optimal path.

If the costs of the arcs satisfy the greedy criterion, it is easy and clear to determine which arc and vector should be chosen. However, when the greedy condition is not satisfied, e.g., the costs of the candidates are all positive, called the undecidable condition, the *branch and bound* (B&B) method is applied by making use of stack or queue architecture.

Example 2. To illustrate the searching process, let us consider the ECG in Fig. 1. According to the searching principle and greedy properties, let us start at the maximum node and select the arc with negative cost, i.e., arc (101). When the undecidable situation occurs at node {101} with the costs of candidates (011) and (110) equaling 0.060 and 0.155, respectively, the greedy algorithm will fail in this situation. If the vector (011) is chosen via the greedy rule, the local optimal basis element {001, 100} with an MSE value equaling 0.682 is obtained by traversing the dotted path. If the B&B algorithm is utilized in the undecidable case, the global optimal basis element {100} with MSE⟨{100}⟩=0.638 will be found. When node {001, 100} is reached, the successors do not need to be examined since the bound condition is satisfied. Therefore, only nine vertices instead of eighteen are traversed and examined by mixing the greedy and B&B algorithms.

III. Constraint Satisfaction Searching Algorithm

In Han and Fan (1994), though the greedy property reduces the time spent traversing of a lot of redundant nodes, and the B&B algorithm guarantees that the optimal solution will be found, a lot of time to traverse a large number of nodes in the worst case and many memory spaces are needed to store the information in the undecidable cases. In this paper, if all the arcs are positive, some inference rules and a constraint satisfaction searching algorithm will be employed to make sure that the chosen candidate vector is in the optimal path. Hence, it can also be guaranteed that the filter we obtain is a global optimal solution, and the redundant nodes will be ignored via the constraint inference rules.

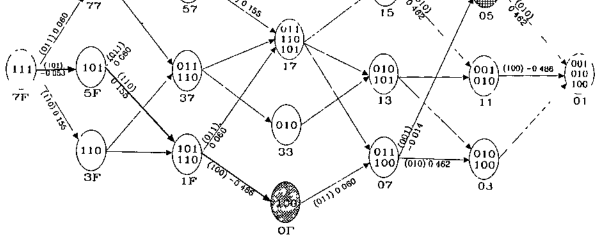


Fig. 1. The error code graph, when N=3.

1. Transformation of ECG to the Basis Graph

Before describing the inference rules formally, let us firstly introduce the *basis graph*, which was described by Dougherty (1992), to explain the following inference rules and illustrations. The basis graph is constructed as an undirected graph with N+1 horizontal levels. For instance, the basis graph for N=3 is constructed as shown in Fig. 2.

Now, let us discuss the generation of a candidate vector set via the basis graph. In the basis graph, the vectors of the current basis $\mathrm{BAS}(\Psi)$ included inside the horizontal-vertical-line circles are a subset of the basis graph. Furthermore, the extension set $\mathrm{EXT}(\mathrm{BAS}(\Psi))$ of the basis $\mathrm{BAS}(\Psi)$ as shown in the dotted area is also a subset of the basis graph. Based on the current basis, the vectors in the candidate vector set $\mathrm{CAN}(\mathrm{BAS}(\Psi))$ as shown inside the circles with horizontal line patterns are generated according to Eq. (3). Furthermore, the vector of the basis graph can be considered as the arc of the ECG, whose cost is displayed below the corresponding vector. While searching for the optimal filter, the basis graph can be adopted to illustrate the searching status of ECG.

The following example demonstrates the status representation of the ECG via the basis graph.

Example 3. Consider the basis element {101, 110} shown in Example 2, which is the element inside the circles with vertical and horizontal line patterns in the ECG. According to the generation rules of the candidate vector set, the candidate vector set CAN({101, 110})={011, 100} drawn inside the horizontal-line-pattern circles is obtained as shown in Fig. 2(a). Due to the greedy property, the vector (100) with the smallest negative cost is chosen as the candidate vector to form the new basis {100} with extension set {100, 110, 101, 111} as shown in Fig. 2(b).

As stated above, the arc in the ECG is a node of the basis graph. The illustrative examples in the rest of this paper will be based on the four-observation filter. However, if N≥4, the ECG is too large and

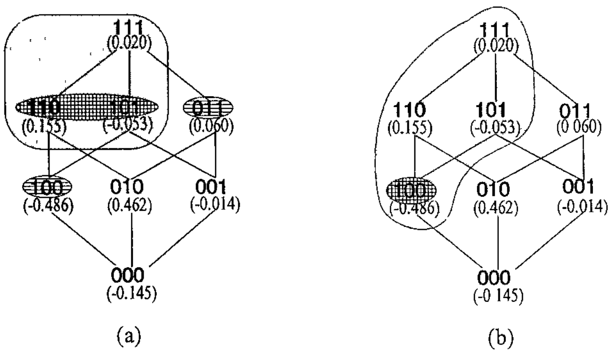


Fig. 2. The generation of a candidate vector set and the representation of the ECG.

complex to be displayed due to space limitations. Therefore, the basis graph is adopted to represent the searching status of the ECG.

2. Constraint Inference Rules

According to the generation rule of the candidate vector set as stated in Eq. (3), a vector $\tilde{u}$ can be chosen as a candidate vector after all the vectors in its extension set EXT($\{\tilde{u}\}$) have been selected. Hence, while searching for the optimal path, if vector $\tilde{u}$ is chosen as an element of the optimal path, we must pay the costs of all the density values of vectors which are in the extension set but have not been traversed. Based on the above observation, the *demanded cost* for vector $\tilde{u}$ in the remaining set is defined as follows:

Definition 1. (DEMANDED COST) Consider the current basis set BAS(Ψ) and a vector $\tilde{u}$, called the demanded vector, in the remaining set U\BAS(Ψ). The demanded cost for this vector $\tilde{u}$ is defined as

$$\mathrm{DC}(\tilde{u}) = \sum \{(-f(\tilde{y},1)+f(\tilde{y},0): \tilde{y} \in \mathrm{EXT}(\tilde{u}) \setminus (\mathrm{EXT}(\tilde{u}) \cap \mathrm{BAS}(\Psi))\}, \quad \textbf{(5)}$$

where $-f(\tilde{0},1)+f(\tilde{0},0)=0$. That is: we have to pay the cost $\mathrm{DC}(\tilde{u})$ before searching for the demanded vector $\tilde{u}$.

In the above definition, the demanded cost $\mathrm{DC}(\tilde{u})$ may be negative or positive. Furthermore, the cost $-f(\tilde{0},1)+f(\tilde{0},0)$ equals zero because the null vector $\tilde{0}$ generates a zero result and never appears in the optimal path of ECG. Note that: the cost of each vector and the demanded cost for each vector for the status representation via the basis graph are displayed as the first and second items below the corresponding vector, respectively.

Now let us begin to describe the constraint satisfaction searching algorithm. While traversing the optimal solution in the ECG, if the undecidable condition of the greedy condition occurs, we choose vector $\tilde{x}$, which is an element of the extension set EXT($\tilde{u}$) of vector $\tilde{u}$ with the minimum demanded cost $\mathrm{DC}(\tilde{u})$. Thus, the principle of the constraint satisfaction searching algorithm is to choose the satisfied candidate vector $\tilde{x}$ to reach vector $\tilde{u}$ with the minimum demanded cost $\mathrm{DC}(\tilde{u})$ as quickly as possible. If two vectors $\tilde{x}$ and $\tilde{y}$ belonging to the candidate vector set CAN(BAS(Ψ)) possess the same minimal demanded cost, then we choose the vector which possesses the second smallest demanded cost as the candidate vector.

Before stating the formal definitions for the inference rules, three set definitions are firstly stated for the later descriptions of the inference rules.

Definition 2. (ASSOCIATED SET) An associated set ASO($\tilde{x}$) for vector $\tilde{x}$ in the candidate vector set, i.e., $\tilde{x} \in$ CAN(BAS(Ψ)), is a collection of vectors that satisfies

$$\mathrm{ASO}(\tilde{x}) = \{\tilde{y} : \tilde{x} \wedge \tilde{y} = \tilde{y}\ , \tilde{y} \in \mathrm{U} \setminus \mathrm{EXT}(\mathrm{BAS}(\Psi))\}. \quad \textbf{(6)}$$

The associated set ASO($\tilde{x}$) for vector $\tilde{x}$ can be considered as a subset of the remaining set U\EXT(BAS(Ψ)). Those vectors belonging to set ASO($\tilde{x}$) can not appear before vector $\tilde{x}$ in any path of the ECG.

Definition 3. (DEMANDED VECTOR SETS/CONSTRAINT CANDIDATE VECTOR SET) The family sets of the demanded vector set and constraint candidate vector set are defined in the recursive form:

$$\mathrm{DVS}(\mathrm{BAS}(\Psi)) = \begin{cases} \mathrm{DVS}_0 = \mathrm{U} \setminus \mathrm{EXT}(\mathrm{BAS}(\Psi)); \\ \mathrm{DVS}_{i+1} = \{\cup_{j=1}^{M} \mathrm{ASO}(\tilde{x}_j)\} \setminus \mathrm{ASO}(\tilde{u}_i), \\ \qquad \text{for } i = 0, 1, 2, \ldots; \end{cases} \quad \textbf{(7)}$$

and

$$\mathrm{CAN}_i^C(\mathrm{BAS}(\Psi), \tilde{u}_i) = \mathrm{CAN}(\mathrm{BAS}(\Psi)) \cap \mathrm{EXT}(\tilde{u}_i), \quad \text{for } i = 0, 1, 2, \ldots, \quad \textbf{(8)}$$

where $\tilde{u}_i \in \mathrm{DVS}_i$, $x_j \in \mathrm{CAN}_i^C(\mathrm{BAS}(\Psi), \tilde{u}_i)$, and the size of the i^{th} constraint candidate vector set is larger than one, i.e., $|\mathrm{CAN}_i^C(\mathrm{BAS}(\Psi), \tilde{u}_i)| = M > 1$. Here, vector $\tilde{u}_i$, called the minimal demanded vector, satisfies

$\text{DC}(\tilde{u}_i) = \min\{\text{DC}(\tilde{v}_k) : \tilde{v}_k \in DVS_i\} < 0$.

In **Definition 3**, the i^{th} constraint candidate vector set $\text{CAN}_i^C(\text{BAS}(\Psi), \tilde{u}_i)$ is defined as the intersection of the candidate vector set CAN(BAS(Ψ)) and the extension set $\text{EXT}(\tilde{u}_i)$ of the minimal demanded vector $\tilde{u}_i$ in the i^{th} demanded vector set DVS_i. These three set definitions are stated with an eye to choosing the candidate vector correctly for the undecidable case of greedy property. The selection criterion is described in the following theorem.

Theorem 1. (INFERENCE RULE I) Suppose the current basis BAS(Ψ) does not satisfy the greedy property while searching the ECG. Consider two vectors $\tilde{u}_i$ and $\tilde{v}_i$ in the i^{th} demanded vector set DVS_i, where $\tilde{u}_i \neq \tilde{v}_i$ and i=0, 1, 2, ... The vector $\tilde{u}_i$ is assumed to be the vector with the smallest negative demanded cost $\text{DC}(\tilde{u}_i)$ in set DVS_i. If vector $\tilde{y}$ belonging to sets $\text{EXT}(\tilde{v}_i)$ and CAN(BAS(Ψ)) but not belonging to set $\text{EXT}(\tilde{u}_i)$, i.e., $\tilde{y} \notin \text{CAN}_i^C(\text{BAS}(\Psi), \tilde{u}_i)$, is in the optimal path, a vector $\tilde{x}$ belonging to the sets $\text{EXT}(\tilde{u}_i)$ and CAN(BAS(Ψ)), i.e., $\tilde{x} \in \text{CAN}_i^C(\text{BAS}(\Psi), \tilde{u}_i)$, is also in the optimal path. That is, the candidate vector of the optimal path is an element of the i^{th} constraint candidate vector set $\text{CAN}_i^C(\text{BAS}(\Psi), \tilde{u}_i)$.

According to the above theorem, we choose those vectors which belong to the 0^{th} constraint candidate vector set $\text{CAN}_0^C(\text{BAS}(\Psi), \tilde{u}_0)$ as the candidate vector to put into the optimal path for the undecidable situation of the greedy property, where vector $\tilde{u}_0$ is the minimal demanded vector with the smallest negative demanded cost $\text{DC}(\tilde{u}_0)$ in the initial demanded vector set DVS_0. If the size of the 0^{th} constraint candidate vector set is larger than one, we generate the next demanded vector set DVS_1 to delete the impossible vectors. The truncating operation is repeated until the size of the i^{th} constraint candidate vector set $\text{CAN}(\text{BAS}(\Psi), \tilde{u}_i)$ is equal to one or until all the demanded costs in set DVS_i are positive. If the former condition is satisfied, we can easily decide which vector is to be put into the optimal path. On the other hand, if the latter condition occurs, the truncating operation via inference rule I will also be terminated. This case will be discussed in inference rule II.

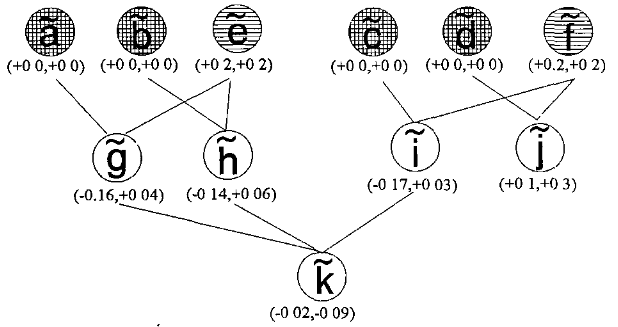


Fig. 3. A contradictory example of Theorem 1.

Next, let us consider the contradiction of inference rule I as shown in Fig. 3. In Fig. 3, vectors $\tilde{a}$, $\tilde{b}$, $\tilde{c}$, and $\tilde{d}$ included inside the horizontal-vertical-line circles are the elements of the current basis. Vectors $\tilde{e}$ and $\tilde{f}$ belonging to the candidate vector set are drawn inside the horizontal line circles. Furthermore, the cost of each vector and the demanded cost for each vector are shown below the corresponding vector. In the first competition, vectors $\tilde{e}$ and $\tilde{f}$ are both winners to be kept for the next competition. In the second competition, since all the demanded costs in the 1^{st} demanded vector set $DVS_1 = \{\tilde{e}, \tilde{f}, \tilde{g}, \tilde{h}, \tilde{i}, \tilde{j}\}$ are positive, the selection process making use of inference rule I will not work well. If vector $\tilde{f}$ is chosen as the candidate vector due to the minimum demanded cost +0.03 in set DVS_1, the optimal solution obtained by selecting vectors $\tilde{e}, \tilde{g}$, and $\tilde{h}$ will be missed. Therefore, another inference rule will be developed and defined in the next definition to tackle this problem.

Definition 4. (MULTI-DEMANDED COST) If all the demanded costs $\text{DC}(\tilde{v}_i)$ in set DVS_i are positive, a multi-demanded cost for a vector $\tilde{u}$ in the i^{th} demanded vector set DVS_i is defined as follows:

$$\text{MDC}(\tilde{u}) = \begin{cases} 0 & \text{if } M = 0; \\ \Sigma\{DC(\tilde{y}) : \tilde{y} \in X_{\tilde{x}}\} - (M-1) \times \text{DC}(\tilde{x}) & \text{if } M = 1, 2, 3, \ldots, \end{cases} \quad \textbf{(9)}$$

where $X_{\tilde{u}} = \{\tilde{y} : -f(\tilde{y}, 1) + f(\tilde{y}, 0) < 0,\ \text{DC}(\tilde{y}) < \text{DC}(\tilde{u}),\ \tilde{y} \wedge \tilde{u} = \tilde{y},\ \text{HAM}(\tilde{u}, \tilde{y}) = 1\}$, and the integer number M is the size of set $X_{\tilde{u}}$. Here, vector $\tilde{u}$ is called the multi-demanded vector in set DVS_i.

For the sake of simplification, the collection of multi-demanded vectors is also called the demanded vector set as in Definition 3. Thus, the definition of the demanded vector set given in Definition 3 will be modified accordingly.

Definition 5. (DEMANDED VECTOR SETS: MODIFIED) The family sets of the demanded vector set are defined in the recursive form:

$\text{DVS}(\text{BAS}(\Psi))$

$$= \begin{cases} \mathrm{DVS}_0 = U \setminus \mathrm{EXT}(\mathrm{B{\small AS}}(\Psi)); \\ \mathrm{DVS}_{i+1} = \{\cup_{j=1}^{M} \mathrm{ASO}(\tilde{x}_j)\} \setminus \mathrm{ASO}(\tilde{u}_i), \\ \qquad \text{for } i = 0, 1, 2, \ldots, \end{cases} \tag{10}$$

where $\tilde{u}_i \in DVS_i$, $\tilde{x}_j \in \mathrm{CAN}_i^C(\mathrm{B{\small AS}}(\Psi), \tilde{u}_i)$, and $|\mathrm{CAN}_i^C(\mathrm{B{\small AS}}(\Psi), \tilde{u}_i)| = M > 1$. The criteria for selecting vector $\tilde{u}_i$ are modified as follows. Consider the minimal demanded vector $\tilde{a}$ whose demanded cost satisfies $\mathrm{DC}(\tilde{a}) = \min\{\mathrm{DC}(\tilde{v}_k): \tilde{v}_k \in \mathrm{DVS}_i\}$, and the multi-demanded vector $\tilde{b}$ whose multi-demanded cost satisfies $\mathrm{MDC}(\tilde{b}) = \min\{\mathrm{MDC}(\tilde{v}_k): \tilde{v}_k \in \mathrm{DVS}_i\}$; the selection of vector $\tilde{u}_i$ is defined as

$$\tilde{u}_i = \begin{cases} \tilde{a} & \text{if } \mathrm{DC}(\tilde{a}) < 0; \\ \tilde{b} & \text{if } \mathrm{DC}(\tilde{a}) \geq 0, \mathrm{MDC}(\tilde{b}) < 0; \\ \text{null} & \text{otherwise}. \end{cases} \tag{11}$$

In addition, the i^{th} constraint candidate vector set is not changed.

From the above definition, when the undecidable case of inference rule I occurs, the multi-demanded cost $\mathrm{MDC}(\tilde{u})$ can be considered as the extra benefit after the vector $\tilde{u}$ has been chosen as an element of the optimal path. If the value $\mathrm{MDC}(\tilde{u})$ is negative, the traversed path will enter the node with an MSE value smaller than that of the current node. On the other hand, if the $\mathrm{MDC}(\tilde{u})$ is positive, the following theorem is adopted to select the candidate vector correctly for the undecidable case of inference rule I.

Theorem 2. (Inference Rule II) If the greedy condition and inference rule I are not satisfied while searching the ECG, i.e., no vector in the candidate vector set is negative and all demanded costs of vectors in the i^{th} demanded vector set DVS_i are positive, then the candidate vector of the optimal path must be an element of the i^{th} constraint candidate vector set $\mathrm{CAN}_i^C(\mathrm{B{\small AS}}(\Psi), \tilde{u}_i)$.

Theorems 1 and 2 are very similar in selecting the candidate vector. The differences between them are the demanded cost and the multi-demanded cost formulations. Theorem 2 solves the undecidable cases of Theorem 1. However, when the undecidable case of Theorem 2 occurs, i.e., all the multi-demanded vectors are positive, the terminated rule must, thus, be developed to terminate the undecidable condition.

Theorem 3. (Inference Rule III: Terminated Rule)) If the undecidable situation occurs and inference rules I and II are not satisfied while searching the ECG, i.e., all the demanded costs and all the multi-demanded costs in the demanded vector set DVS_i are positive, then the ECG will satisfy the following rules:

(1) If i=0, there is no element in the remaining set belonging to the optimal path.

(2) If $i \geq 1$, all the vectors in the i^{th} constraint candidate vector set are in the optimal path.

From Theorems 1, 2, and 3, if the undecidable condition of the greedy property occurs while traversing the ECG, we will be able to select the candidate vector correctly by making use of these three inference rules.

3. Constraint Satisfaction Searching Algorithm

According to the description in the previous subsection, the constraint satisfaction searching algorithm used in searching for the optimal path of the optimal filter can be formally described as follows.

Algorithm

Step 1: Compute the density values of $f(x[1], x[2], \ldots, z)$. There are 2^{N+1} items for N observation random variables and an estimated variable Z.

Step 2: Initialize some values. Assign the starting basis $\{(\overbrace{11 \ldots 1}^{N})\}$ to the current basis element $\mathrm{B{\small AS}}(\Psi)$ and $(\overbrace{0, 1, 1, \ldots, 1}^{2^N})$ with value $2^{2^N-1}-1$ to the current error code vector $\tilde{ECV}$. Then, compute the MSE value of the current basis element and assign the current MSE value to that of the optimal basis element. Moreover, compute the demand cost and multi-demanded cost for each vector in the basis graph.

Step 3: Determine the candidate vector set $\mathrm{CAN}(\mathrm{B{\small AS}}(\Psi))$ from the current basis element and the error code vector to choose the candidate vector for transiting the vertices correctly.

Step 4: If the terminated condition is satisfied, terminate the constraint satisfaction searching procedure. Otherwise, go to Step 5.

Step 5: Choose the candidate vector of the optimal path from the candidate vector set.

5.1: (Greedy property) If one of the vectors in the candidate vector set is negative, utilize the *greedy* algorithm to choose the vector with the smallest negative cost as the candidate vector $\tilde{x}$. Ignore the others, and then go to Step 6. Otherwise, go to Step 5.2.

5.2: Initialize i=0.

5.3: (Inference rule I) Assume vector $\tilde{u}_i$ is the minimal demanded vector. If one of the demanded costs in the demanded vector set DVS_i is negative, delete the impossible vectors which do not belong to the i^{th} constraint candidate vector set $CAN_i^C(\text{BAS}(\Psi), \tilde{u}_i)$ repeatedly. If the size of the i^{th} constraint candidate vector set is equal to one, go to Step 6; otherwise, go to Step 5.4.

5.4: (INFERENCE RULE II) Assume vector $\tilde{u}_i$ is the minimal multi-demanded vector. If one of the multi-demanded costs in the demanded vector set DVS_i is negative, delete the impossible vectors which do not belong to the i^{th} constraint candidate vector set $CAN_i^C(\text{BAS}(\Psi), \tilde{u}_i)$ repeatedly. If the size of the i^{th} constraint candidate vector set is equal to one, goto Step 6; otherwise, go to Step 5.5.

5.5: (INFERENCE RULE III) If i=0, the terminated condition is satisfied. Terminate the constraint satisfaction searching algorithm. Otherwise, if $i \geq 1$, choose any vector in the i^{th} constraint candidate vector set $CAN_i^C(\text{BAS}(\Psi), \tilde{u}_i)$ as the candidate vector.

Step 6: Add the candidate vector $\tilde{x}$ to the current basis, and set the $\tilde{x}^{th}$ bit of $E\tilde{C}V$ to '0' according to the definition of the candidate vector set. Then, add the value $-f(\tilde{x},1)+f(\tilde{x},0)$ to the MSE value of the current basis. In addition, update the demanded cost and multi-demanded cost tables accordingly.

Step 7: Compare the new MSE value with the optimal MSE value. If the new value is smaller than the optimal MSE value, update the optimal MSE value with that of the new value, and set the optimal basis element to the new basis element.

Step 8: Update the current basis element and MSE value with the new basis element and new MSE value, respectively.

Step 9: Repeat Step 3 to Step 8.

4. Illustrative Example

An illustrative example is given in this subsection to demonstrate the complete process of the constraint satisfaction searching algorithm. During the searching process, the searching status of the ECG for a four-observation filter is that shown in Fig. 4. Firstly, the cost of each vector (arc) is shown below the corresponding vector of Fig. 4(a). In Fig. 4(a), the sequential orders of the chosen vectors via the greedy algorithm are written in the upper-right corner. The local optimal filter {0011, 1010} is obtained by making use of the greedy method. From Fig. 4(b)-(e), the undecidable cases of the greedy property will be given and solved by using three inference rules.

In Fig. 4(b), the undecidable case of the greedy condition occurs at node $\text{BAS}(\Psi)$={0011}, where CAN({0011})={1100, 1101}. According to the definitions of the demanded vector set and the constraint candidate vector set, the minimal demanded vector (1010) with the negative demanded cost –0.0068 generates the constraint candidate vector set CAN_0^C({0011}, {1010})={1110}. Since the size of set CAN_0^C({0011}, {1010}) is equal to one, vector (1110) is chosen to join into the optimal path.

After joining vector (1110) into the optimal path, vector (1010) is then selected as the candidate vector to obtain the new basis element {0011, 1010} as shown in Fig. 4(c). At this time, since both vectors (0110) and (1101) in set CAN({1010, 0011}) and all the demanded costs in set DVS_0 are positive, the multi-demanded cost function and inference rule II are adopted to facilitate selection of the candidate vector. In Fig. 4(c), the multi-demanded cost for each vector is shown as the third item below the corresponding vector.

Based on Theorem 2 (inference rule II), the multi-demanded vector (1100) with the smallest negative multi-demanded cost -0.2096 is obtained to generate the constraint candidate vector set {1101}. Since the size of the constraint candidate vector set is equal to one, vector (1101) is thereby selected as an element of the optimal path. After the joining operation, vector (0101) is next chosen to form the basis element {0011, 0101, 1010} as shown in Fig. 4(d).

In Fig. 4(d), the candidate vector set {0110, 1001, 1100} for the current basis element {0011, 0101, 1010} is obtained to select the candidate vector. In the first competition, since vector (1001) does not belong to the 0^{th} constraint candidate vector set, it is deleted by applying inference rule I. In the second competition, since all the demanded costs in set DVS_1 are positive, the multi-demanded cost function is employed to construct the 1^{st} constraint candidate vector set CAN_1^C({0011, 0101, 1010}, (1100))={1100} according to the modified definition of the demanded vector set. Applying inference rule II, we choose vector (1100) to form the new basis element {0011, 0101, 1010, 1100}.

Two vectors (1101) and (0110) are then chosen as the candidate vectors based on inference rule I. When the basis element becomes {0011, 0100, 1000} as shown in Fig. 4(e), the terminated rule (inference rule III) is satisfied. We terminate the constraint satisfaction searching algorithm. Hence, the optimal filter {0011, 0100, 1000} is found.

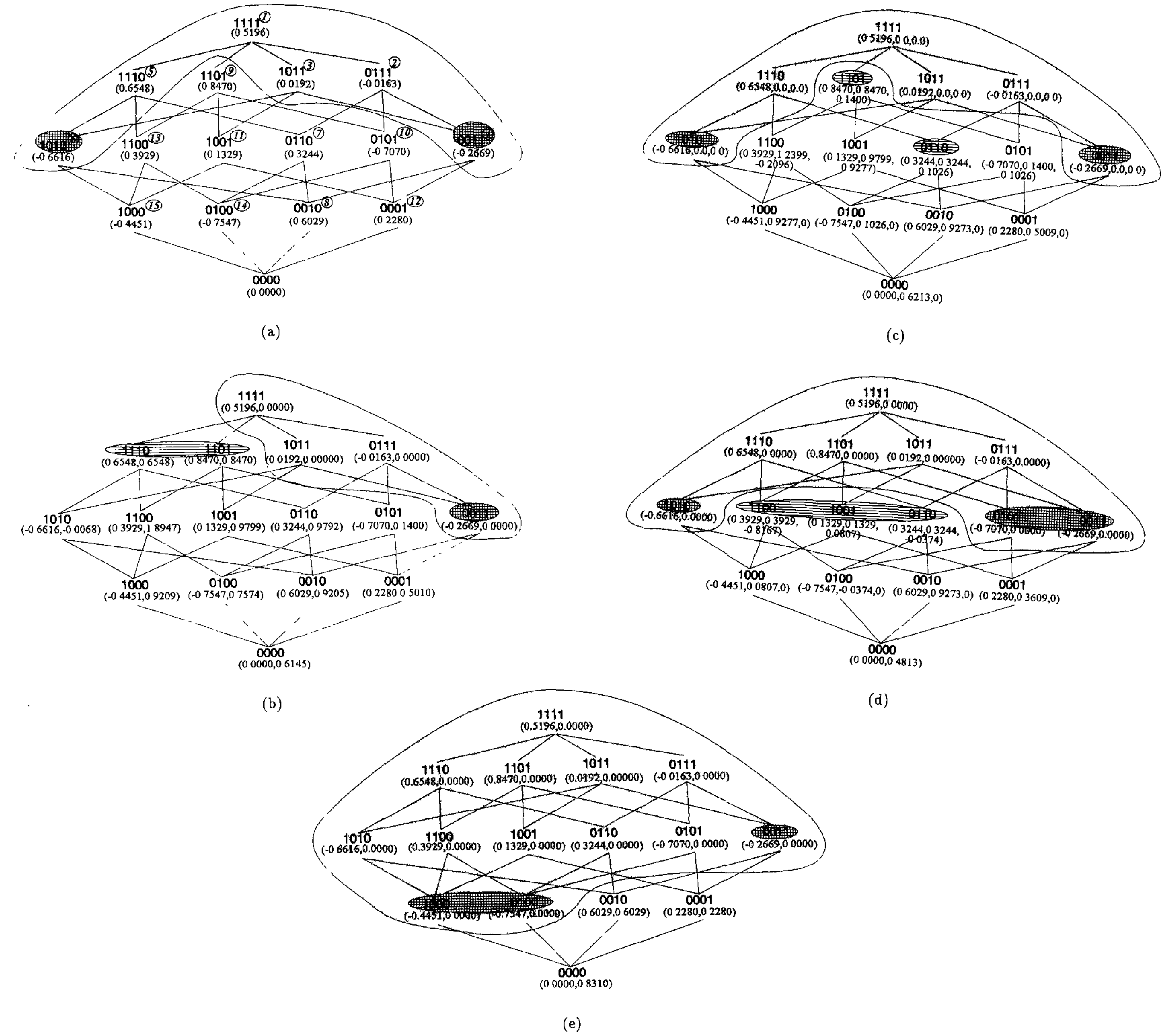


Fig. 4. The complete searching process of the constraint satisfaction searching algorithm.

IV. Experimental Results and Discussion

The proposed method has been implemented using C language on an IBM PC-486. Experiments were conducted to show the validity of designing the optimal morphological erosion filter on a personal computer. The probability value tables were simulated to test the performance of our proposed algorithm. In addition to evaluation of the searching time, the effect generated by the optimal erosion filter was also examined. The time complexity and memory requirement of our proposed method will be discussed in subsection IV. 2.

1. Experimental Results

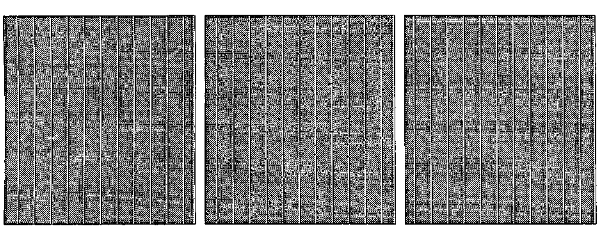
The performance of our proposed method was evaluated by analyzing the searching time required to find the optimal solution. 10,000 probability value tables for various observation sizes were randomly generated and tested by searching the entire space using our proposed method. Shown in Table 3 is the execu-

Table 3. The Execution Time for Various Random Variables

N	1	2	3	4	5	6
Traditional Method	–	–	0.002741 second	0.053915 second	6.16577 second	>6 hours
constraint satisfaction Method	–	–	0.002993 second	0.008634 second	0.02968 second	0.1159 second
Greedy and B&B Method	–	–	0.006684 second	0.025630 second	0.43010 second	180 second

tion time taken in finding the optimal basis element combination. When the number of observation random variables N was larger than six, the searching time over the entire space was more than six hours. Thus, it seems impractical to implement it on a personal computer. Hence, the estimation times for $N \geq 7$ are not listed in this table. From Table 3, we can find that the execution time used in searching the whole space far exceeded that of our proposed method.

In addition to performance analysis of the searching time, the results generated by the estimated morphological erosion filter obtained using our proposed method are also presented. Two binary images corrupted by noise are shown in Figs. 5 and 6. The size of the structuring element for both experimental examples was chosen to be 3×3 with the center pixel as the origin. The ideal images of these experimental examples are displayed in Figs. 5(a) and 6(a) in 100 and 300 dpi resolutions with the input images to be processed shown in Figs. 5(b) and 6(b), respectively.



(a) (b) (c)

Fig. 5. Experimental result 1.

(a) (b) (c)

Fig. 6. Experimental result 2.

Shown in Figs. 5(c) and 6(c) are the resulting images processed via the structuring elements which were estimated by our proposed algorithm. In experiment 1, the optimal filter was composed of ten structuring elements, where the MSE value was 5.486071×10^{-4}. In experiment 2, 100 structuring elements constituting the optimal filter were employed to generate the resulting image with MSE=5.702533×10^{-4}. The searching time taken in finding the optimal filter for experiments 1 and 2 was 4 and 13 seconds, respectively.

2. Time Complexity Analysis and Memory Requirement

The time complexity analysis of the constraint satisfaction searching algorithm is discussed below. As mentioned by Han and Fan (1994), we need $O(2^{2N})$ to generate the candidate vector set CAN(BAS(Ψ)) for each traversed node BAS(Ψ). The searching time needed to find the optimal filter is $O(m \cdot 2^N)$, where m is the number of traversed nodes. The value m will be very large in the worst case. However, the time needed to obtain the candidate vector set CAN(BAS(Ψ)) is $O(N \cdot 2^N)$. In order to test whether vector $\tilde{x}$ with n '1' and $N-n$ '0' digits is an element of the candidate vector set, only $N-n$ vectors at level $n+1$ in the basis graph need to be checked. If these $N-n$ vectors are all in set EXT(BAS(Ψ)), then vector $\tilde{x}$ is said to be an element of the candidate vector set. Thus, $O(N)$ is needed to test whether vector $\tilde{x}$ belongs to the candidate vector set and $O(N \cdot 2^N)$ to generate the candidate vector set CAN(BAS(Ψ)). Due to the morphological filter criteria, there are at most $C\left(N, \left\lfloor \frac{N}{2} \right\rfloor\right)$ vectors in the candidate vector set. If these vectors in the candidate vector set satisfy the greedy property, we need $O\left(C\left(N, \left\lfloor \frac{N}{2} \right\rfloor\right)\right)$ to decide the candidate vector. If the greedy condition is not satisfied, we need $O(2^N)$ to truncate the impossible vectors from set CAN(BAS(Ψ)) by making use of inference rules each time. Hence, the time needed is $O\left(C\left(N, \left\lfloor \frac{N}{2} \right\rfloor\right) \cdot 2^N\right)$

to obtain the correct candidate vector of the optimal path. In Han and Fan (1994), the error code graph is constructed as a directed graph with 2^N-1 levels. Therefore, the total time needed by our proposed graphic search-hased method is $O\left(c_1 \cdot 2^N \cdot N \cdot 2^N + c_2 \cdot C\left(N, \left\lfloor \frac{N}{2} \right\rfloor\right) \cdot 2^N \cdot 2^N\right)$, i.e., $O\left(C\left(N, \left\lfloor \frac{N}{2} \right\rfloor\right) \cdot 2^{2N}\right)$, where c_1 and c_2 are two arbitrary positive values.

The memory requirement for our proposed method is two 2^{2^N} bits to represent the error code value $E\hat{C}V$. One is for the optimal filter, and the other is for the current filter. From the error code value of optimal filter ECV_{optimal}, the basis element of the optimal filter can be obtained via the reduction operation. Furthermore, we need $O(2 \cdot 2^N)$ to store the demanded cost and the multi-demanded cost tables for each vector in the basis graph.

V. Conclusion

A novel graphic search-based optimal morphological filter finding algorithm which avoids examination of all possible basis element combinations has been proposed in this paper. The problem of searching for the optimal filter in an extremely large number of structuring element combinations is reduced to the problem of searching for the shortest path in the ECG. The greedy property and inference rules help us choose the candidate vector correctly during the searching process of the optimal path. With these graphic searching techniques, there are at most 2^N-1 nodes to be traversed in the ECG. Experiments have been conducted to verify our proposed method. Experimental results reveal the validity and efficiency of the proposed method in finding the optimal binary morphological erosion filter.

Appendix

Before the proof of the theorems, a lemma is first described to facilitate the proof of later theorems.

Lemma 1. Consider the current basis set $\text{BAS}(\Psi)$ and a vector $\tilde{u}$ in the remaindered set $U \backslash \text{BAS}(\Psi)$. The intersection set of extension set $\text{EXT}(\tilde{u})$ for vector $\tilde{u}$ and candidate vector set $\text{CAN}(\text{BAS}(\Psi))$ is a collection of $X = \{\tilde{x} : \tilde{x} \in (\text{EXT}(\tilde{u}) \cap \text{CAN}(\text{BAS}(\Psi)))\}$. If all the vectors in set X appear behind vector $\tilde{y}_n$ of a sub-path ($\tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_n$), all the vectors in extension set $Y = \text{EXT}(\tilde{u}) \backslash (\text{EXT}(\tilde{u}) \cap \text{EXT}(\text{BAS}(\Psi)))$ for vector $\tilde{u}$ are also behind vector $\tilde{y}_n$.

Proof: Assume that a vector $\tilde{y}_i$ belongs to extension set Y and appears before vector $\tilde{y}_n$. Since all the vectors in set X are behind vector $\tilde{y}_n$, vector $\tilde{y}_i$ is not in set X, and a vector $\tilde{x}$ in set X belongs to set $\text{EXT}(\tilde{y}_i)$, i.e., $\tilde{x} \in \text{EXT}(\tilde{y}_i)$.

Since vector $\tilde{y}_i$ is a candidate vector for basis set $\text{BAS}_{y_i-1}(\Psi)$, i.e., $\tilde{y}_i \in \text{CAN}(\text{BAS}_{y_i-1}(\Psi))$, it satisfies the candidate vector set condition:

$$\tilde{y}_i \in U \backslash \text{EXT}(\text{BAS}(\Psi)), \text{ and } \text{EXT}(\tilde{y}_i) \backslash \{\tilde{y}_i\} \subseteq \text{EXT}(\text{BAS}_{y_i-1}(\Psi)).$$

Therefore,

$$\tilde{x} \in \text{EXT}(\tilde{y}_i) \backslash \{\tilde{y}_i\} \subseteq \text{EXT}(\text{BAS}_{y_i-1}(\Psi)),$$

and vector $\tilde{x}$ is a candidate vector of node $\text{BAS}_{y_i-1}(\Psi)$.

Since all the vectors in set X are behind vector $\tilde{y}_n$, they will not be chosen when the basis set is $\text{BAS}_{y_i-1}(\Psi)$. Hence, all the vectors in set X do not belong to set $\text{EXT}(\text{BAS}_{y_i-1}(\Psi))$. This assertion that vector $\tilde{y}_i$ is an element of set $\text{EXT}(\tilde{u})$ is a contradiction. Thus, any vector $\tilde{y}_i$ is not a vector of set Y, and all the vectors in set Y are behind vector $\tilde{y}_n$.

Theorem 1. (INFERENCE RULE I) Suppose the current basis $\text{BAS}(\Psi)$ does not satisfy the greedy property while searching the ECG. Consider two vectors $\tilde{u}_i$ and $\tilde{v}_i$ in the i^{th} demanded vector set DVS_i, where $\tilde{u}_i \neq \tilde{v}_i$ and i=0, 1, 2, ... Vector $\tilde{u}_i$ is assumed to be the vector with the smallest negative demanded cost $\text{DC}(\tilde{u}_i)$ in set DVS_i. If vector $\tilde{y}$ belonging to sets $\text{EXT}(\tilde{v}_i)$ and $\text{CAN}(\text{BAS}(\Psi))$ but not belonging to set $\text{EXT}(\tilde{v}_i)$, i.e., $\tilde{y} \notin \text{CAN}_i^C(\text{BAS}(\Psi), \tilde{u}_i)$, is in the optimal path, then a vector $\tilde{x}$ belonging to sets $\text{EXT}(\tilde{u}_i)$ and $\text{CAN}(\text{BAS}(\Psi))$, i.e., $\tilde{x} \in \text{CAN}_i^C(\text{BAS}(\Psi), \tilde{u}_i)$, is also in the optimal path. That is, the candidate vector of the optimal path is an element of the i^{th} constraint candidate vector set $\text{CAN}_i^C(\text{BAS}(\Psi), \tilde{u}_i)$.

Proof: In this proof, the contradictory techniques are used. Since vector $\tilde{y}$ is in the optimal path, the optimal sub-path ($\tilde{y} = \tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_n$) after the current node $\text{BAS}(\Psi)$ can be found.

Suppose no vector in the i^{th} constraint candidate vector set $\text{CAN}_i^C(\text{BAS}(\Psi), \tilde{u}_i)$ is located in the optimal path. Thus, all the vectors in set $\text{CAN}_i^C(\text{BAS}(\Psi), \tilde{u}_i)$ must appear behind vector $\tilde{y}_n$, which is the last vector in the optimal path. As a consequence of Lemma 1, any vector $\tilde{x}_i$ in set $\text{EXT}(\tilde{u}_i) \backslash (\text{EXT}(\tilde{u}_i) \cap \text{EXT}(\text{BAS}(\Psi))) = \cup_{i=1}^{m} \tilde{x}_i$ can not appear in the optimal sub-path, i.e., $\tilde{x}_i \neq \tilde{y}_j$; i=1, 2, ..., m; j=1, 2, ..., n. Furthermore, we can find another sub-path consisting of a sub-path ($\tilde{x} = \tilde{x}_1, \tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_n$) (Han and Fan, 1994) and a consecutive sub-path ($\tilde{x}_2, \tilde{x}_3, \ldots, \tilde{x}_m = \tilde{u}_i$), where $\text{DC}(\tilde{u}_i) = \Sigma_{i=1}^{m}(-f(\tilde{x}_i, 1) + f(\tilde{x}_i, 0)) < 0$.

As stated above, the MSE values of optimal node $\text{BAS}_{y_n}(\Psi)$ via the optimal path and the value of node $\text{BAS}_{u_i}(\Psi)$ via the new sub-path ($\tilde{x} = \tilde{x}_1, \tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_n, \tilde{x}_2, \ldots, \tilde{x}_m = \tilde{u}_i$) can be calculated as follows:

$$\begin{aligned}
&\text{MSE}\langle \text{BAS}_{yn}(\Psi) \rangle \\
&= \text{MSE}\langle \text{BAS}(\Psi) \rangle + \text{the cost of path}(\tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_n) \\
&= \text{MSE}\langle \text{BAS}(\Psi) \rangle + \sum_{i=1}^{n}(-f(\tilde{y}_i, 1) + f(\tilde{y}_i, 0)); \\
&= \text{MSE}\langle \text{BAS}_{\text{optimal}}(\Psi) \rangle \\
&\text{MSE}\langle \text{BAS}_{u_i}(\Psi) \rangle \\
&= \text{MSE}\langle \text{BAS}(\Psi) \rangle \\
&\quad + \text{the cost of path}(\tilde{x} = \tilde{x}_1, \tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_n, \tilde{x}_2, \ldots, \tilde{x}_m = \tilde{u}_i); \\
&= \text{MSE}\langle \text{BAS}(\Psi) \rangle + (-f(\tilde{x}_i, 1) + f(\tilde{x}_i, 0) \\
&\quad + \sum_{i=1}^{n}(-f(\tilde{y}_i, 1) + f(\tilde{y}_i, 0)) + \sum_{i=2}^{m}(-f(\tilde{x}_i, 1) + f(\tilde{x}_i, 0))
\end{aligned}$$

$$= \mathrm{MSE}\langle \mathrm{BAS}_{\mathrm{optimal}}(\Psi)\rangle + \mathrm{DC}(\tilde{u}_i)$$

$$< \mathrm{MSE}\langle \mathrm{BAS}_{\mathrm{optimal}}(\Psi)\rangle .$$

This assertion that any vector $\tilde{x}$ in set $\mathrm{CAN}_i^C(\mathrm{BAS}(\Psi), \tilde{u}_i)$ is not an element of the optimal path is a contradiction. Thus, there exists a vector $\tilde{x}$ in set $\mathrm{CAN}_i^C(\mathrm{BAS}(\Psi), \tilde{u}_i)$ belonging to the optimal path.

Theorem 2. (Inference Rule II) If the greedy condition and inference rule I are not satisfied while searching the ECG, i.e., no vector in the candidate vector set is negative, and all the demanded costs of the vectors in the i^{th} demanded vector set DVS_i are positive, then the candidate vector of the optimal path must be an element of the i^{th} constraint candidate vector set $\mathrm{CAN}_i^C(\mathrm{BAS}(\Psi), \tilde{u}_i)$.

Proof: Suppose that vector $\tilde{x}$ belonging to the i^{th} constraint candidate vector set $\mathrm{CAN}_i^C(\mathrm{BAS}(\Psi), \tilde{u}_i) = \cup_{j=1}^{m} \tilde{x}_j$ is not an element of the optimal sub-path $(\tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_n)$. According to the results in Han and Fan (1994), there exists a sub-path $(\tilde{x} = \tilde{x}_1, \tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_n, \tilde{x}_2, \tilde{x}_3, \ldots, \tilde{x}_m)$ which is composed of the vector $\tilde{x}$, the optimal sub-path $(\tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_n)$, and the vectors in the i^{th} constraint candidate vector set $\mathrm{CAN}_i^C(\mathrm{BAS}(\Psi), \tilde{u}_i)$.

Assume that set $X_{\tilde{u}_i}$ is the collection of vectors whose costs constitute the multi-demanded cost $\mathrm{MDC}(\tilde{u}_i)$, i.e., $X_{\tilde{u}_i} = \{\tilde{v} : -f(\tilde{v}, 1) + f(\tilde{v}, 0) < 0,\ \mathrm{DC}(\tilde{v}) < \mathrm{DC}(\tilde{u}_i),\ \tilde{v} \wedge \tilde{u}_i = \tilde{v},\ \mathrm{HAM}(\tilde{u}_i, \tilde{v}) = 1\} = \cup_{k=1}^{s} \tilde{v}_k$.

Since $\tilde{v}_j \cup \tilde{v}_k = \tilde{u}_i$, for $j,k = 1, 2, \ldots, s$, $j \neq k$, $\mathrm{EXT}(\tilde{v}_j) \cup \mathrm{EXT}(\tilde{v}_k) = \mathrm{EXT}(\tilde{u}_i)$. Build up a sub-path $(\tilde{x}_{m+1}, \tilde{x}_{m+2}, \ldots, \tilde{x}_n = \tilde{u}_i, \tilde{v}_{11}, \tilde{v}_{12}, \ldots, \tilde{v}_1, \tilde{v}_{21}, \tilde{v}_{22}, \ldots, \tilde{v}_2, \ldots, \tilde{v}_{s1}, \tilde{v}_{s2}, \ldots, \tilde{v}_s)$ after sub-path $(\tilde{x} = \tilde{x}_1, \tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_n, \tilde{x}_2, \tilde{x}_3, \ldots, \tilde{x}_m)$.
Now, let consider the MSE values of nodes $\mathrm{BAS}_{y_n}(\Psi)$ and $\mathrm{BAS}_{v_s}(\Psi)$:

$$\mathrm{MSE}\langle \mathrm{BAS}_{y_n}(\Psi)\rangle$$

$$= \mathrm{MSE}\langle \mathrm{BAS}(\Psi)\rangle + \text{the cost of path}(\tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_n)$$

$$= \mathrm{MSE}\langle \mathrm{BAS}(\Psi)\rangle + \sum_{i=1}^{n} (-f(\tilde{y}_i, 1) + f(\tilde{y}_i, 0))$$

$$= \mathrm{MSE}\langle \mathrm{BAS}_{\mathrm{optimal}}(\Psi)\rangle ;$$

$$\mathrm{MSE}\langle \mathrm{BAS}_{v_s}(\Psi)\rangle$$

$$= \mathrm{MSE}\langle \mathrm{BAS}(\Psi)\rangle$$

$$+ \text{the cost of path}(\tilde{x} = \tilde{x}_1, \tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_n, \tilde{x}_2, \ldots, \tilde{x}_m)$$

$$+ \text{the cost of path}(\tilde{x}_{m+1}, \tilde{x}_{m+2}, \ldots, \tilde{x}_n = \tilde{u}_i) .$$

$$+ \text{the cost of path}(\tilde{v}_{11}, \tilde{v}_{12}, \ldots, \tilde{v}_1)$$

$$\vdots$$

$$+ \text{the cost of path}(\tilde{v}_{s1}, \tilde{v}_{s2}, \ldots, \tilde{v}_s)$$

$$= \mathrm{MSE}\langle \mathrm{BAS}(\Psi)\rangle + \sum_{i=1}^{n} (-f(\tilde{y}_i, 1) + f(\tilde{y}_i, 0)) + \mathrm{DC}(\tilde{u}_i)$$

$$+ \mathrm{DC}(\tilde{v}_1) - \mathrm{DC}(\tilde{u}_i) + \mathrm{DC}(\tilde{v}_2) - \mathrm{DC}(\tilde{u}_i) + \ldots + \mathrm{DC}(\tilde{v}_s) - \mathrm{DC}(\tilde{u}_i)$$

$$= \mathrm{MSE}\langle \mathrm{BAS}_{\mathrm{optimal}}(\Psi)\rangle + \mathrm{MDC}(\tilde{u}_i)$$

$$< \mathrm{MSE}\langle \mathrm{BAS}_{\mathrm{optimal}}(\Psi)\rangle .$$

This assertion that any vector $\tilde{x}$ in set $\mathrm{CAN}_i^C(\mathrm{BAS}(\Psi), \tilde{u}_i)$ is not an element of the optimal path is a contradiction. Thus, there exists a vector $\tilde{x}$ in set $\mathrm{CAN}_i^C(\mathrm{BAS}(\Psi), \tilde{u}_i)$ belonging to the optimal path.

Theorem 3. (Inference Rule III: Terminated Rule) If the undecidable situation occurs, and inference rules I and II are not satisfied while searching the ECG, i.e., all the demanded costs and all the multi-demanded costs in the demanded vector set DVS_i are positive, then the ECG will satisfy the following rules:

(1) If i=0, then there is no element in the remained set belonging to the optimal path.
(2) If i≥1, then all the vectors in the i^{th} constraint candidate vector set are in the optimal path.

Proof:

(1) Assume that the path $(\tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_n)$ is the sub-path of the optimal path after the current node $\mathrm{BAS}(\Psi)$ whose cost is $\sum_{i=1}^{n} \tilde{y}_i < 0$. These vectors $\tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_n$ can be reduced to a set $X = \{\tilde{y}_1', \tilde{y}_2', \ldots, \tilde{y}_m'\}$ which satisfies the morphological basis criteria, and each of them can be the last element of the optimal path.
The induction and contradictory techniques are employed to prove that the summation $\sum_{i=1}^{n} \tilde{y}_i$ is larger than zero.
Basis: If the size of set X equals one, then the summation of vectors $\tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_n$ is equal to $\mathrm{DC}(\tilde{y}_1')$, which is larger than zero.
Induction: Suppose that the hypothesis is true when the size of X equals m or less. That is, $\sum_{i=1}^{n} \tilde{y}_i = \mathrm{DC}(\tilde{y}_1', \tilde{y}_2', \ldots, \tilde{y}_m') > 0$, where $\mathrm{DC}(\tilde{y}_1', \tilde{y}_2', \ldots, \tilde{y}_m')$ is the summation of the costs of those vectors in extension sets $\mathrm{EXT}(\tilde{y}_1')$, $\mathrm{EXT}(\tilde{y}_2')$, ... $\mathrm{EXT}(\tilde{y}')$. If the size of the reduced set X equals m+1, the summation $\sum_{i=1}^{n} \tilde{y}_i$ can be computed as

$$\sum_{i=1}^{n} \tilde{y}_i = \mathrm{DC}(\tilde{y}_1', \tilde{y}_2', \ldots, \tilde{y}_m', \tilde{y}_{m+1}')$$

$$= \mathrm{DC}(\tilde{y}_1', \tilde{y}_2', \ldots, \tilde{y}_m') + \mathrm{DC}(\tilde{y}_{m+1}')$$

$$- \mathrm{DC}(\tilde{y}_1' \cup \tilde{y}_{m+1}', \tilde{y}_2' \cup \tilde{y}_{m+1}', \ldots, \tilde{y}_m' \cup \tilde{y}_{m+1}') .$$

Since all the demanded costs and multi-demanded costs are positive, $\mathrm{DC}(\tilde{y}_{m+1}') > 0$. Furthermore, if the collection of $\mathrm{EXT}(\tilde{y}_1' \cup \tilde{y}_{m+1}')$, $\mathrm{EXT}(\tilde{y}_2' \cup \tilde{y}_{m+1}'), \ldots, \mathrm{EXT}(\tilde{y}_m' \cup \tilde{y}_{m+1}')$ is the subset of collection $\mathrm{EXT}(\tilde{y}_1')$, $\mathrm{EXT}(\tilde{y}_2'), \ldots, \mathrm{EXT}(\tilde{y}_m')$, then $\mathrm{DC}(\tilde{y}_1', \tilde{y}_2', \ldots, \tilde{y}_m') > \mathrm{DC}(\tilde{y}_1' \cup \tilde{y}_{m+1}', \tilde{y}_2' \cup \tilde{y}_{m+1}', \ldots, \tilde{y}_m' \cup \tilde{y}_{m+1}')$. Therefore, the summation of vectors $\tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_n$ is larger than zero. This contradicts the assumption.
From the above proof, we can conclude that sub-path $(\tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_n)$ does not exist in the optimal path. That is, any combination of vectors in set DVS_0 is not the sub-path of the optimal path.

(2) From **Theorems 1** and **2**, there exists a vector in the $(i\text{-}1)^{\text{th}}$ constraint candidate vector set which is a vector of the optimal path. From the conclusion of the above proof, the summation cost of any vector combination in DVS_i is larger than zero. To reach the optimal node, the minimal demanded vector $\tilde{u}_{i-1}$ in set DVS_{i-1} must be traversed. Based on this condition, all the vectors in the i^{th} candidate vector set $\mathrm{CAN}_i^C(\mathrm{BAS}(\Psi), \tilde{u}_i)$ must be selected.

References

Dougherty, E. R. and C. R. Giardina (1988) *Morphological Methods in Image and Signal Processing*. Prentice-Hall, Englewood Cliffs, NJ, U.S.A.

Dougherty, E.R., A. Mathew, and V. Swarnaker (1991) A conditional-expectation-based implementation of the optimal mean-square binary morphological filter. *Proc. SPIE*, **1451**, 137-146.

Dougherty, E. R. and R. M, Haralick (1991) The hole spectrum-model-based optimalization of morphological filters. *Proc. SPIE, Image Algebra and Marphological Image Processing II*, **1568**, 224-232.

Dougherty, E.R. (1992) Optimal mean-square N-observation digital morphological filters. Part I: Optimal binary filters. *CVGIP: Image Understanding*, **55**, 36-54.

Dougherty, E. R. and R. P. Loce (1993) Optimal mean-absolute-error hit-or-miss filters: morphological representation and estimation of the binary condition expectation *Optical Engineering*, **32**, 815-827.

Han, C. C. and K. C. Fan (1994) Finding of optimal binary morphological erosion filter via greedy and branch & bound searching *J. Mathematical Imaging and Vision*, **3**(3), 298-312.

Loce, R. P. and E. R. Dougherty (1991) Using structuring-element libraries to design suboptimal morphological filters. *Proc. SPIE, Image Algebra and Morphological Image Processing II*, **1568**, 233-246.

Loce, R. P. and E. R. Dougherty (1992a) Optimal morphological restoration: The morphological filter mean-absolute-error theorem. *J. Visual Comm. Image Representation*, **3**, 412-432.

Loce, R. P. and E. R. Dougherty (1992b) Facilitation of optimal binary morphological filter design via structuring element libraries and design constraints. *Optical Engineering,* **31**, 1008-1025.

Maragos, P. and R. S. Schafer (1987) Morphological filters, I. Their set-theoretic analysis and relations to linear shift-invariant filters. *IEEE Trans. Acoust. Speech Signal Process*, **35**, 1153-1169.

Maragos, P. (1989) Arepresentation theory for morphological image and signal processing. *IEEE Trans. Pattern Anal. Mach. Intelligence*, **11**, 586-599.

Mathew, A. V., E. R. Dougherty, and V. Swarnakar (1993) Efficient derivation of the optimal mean-square binary morphological filter from the conditional expectation via switching algorithm for discrete power-set lattice. *Circuit System Signal Process*, **12**, 409-430.

Rich, E. (1983) *Artificial Intelligence.* Mcgraw-Hill, New York, NY, U.S.A.

Serra, J. (1988) *Image Analysis and Mathematical Morphology,* Academic Press, New York, NY, U.S.A.

利用貪婪及條件滿足搜尋演算法找尋最佳二位元形態濾波器

韓欽銓*　范國清**

*中央研究院資訊科學研究所

**中央大學資訊工程研究所

摘　要

在本篇文章中，我們提出一種圖形搜尋演算法，找尋最佳二位元形態濾波器。傳統上，我們需要大量的時間來搜尋最佳解，在本文中，搜尋最佳解的問題被轉換成：在錯誤碼圖形中找尋一條具有最短路徑的途徑。並利用兩種圖形搜尋方法：貪婪及條件滿足搜尋演算法，避免在大量的搜尋空間中，搜尋最佳解，實驗結果證實了我們的方法是非常不錯的。