

逻辑列阵实现的数学形态学 双值像处理器*

王斌泉 刘立人 梁 丰

(中国科学院上海光学精密机械研究所, 上海 201800)

摘 要 从逻辑操作的角度出发, 研究了用逻辑列阵来实现数学形态学像操作的可能性, 并给出了几种基本的数学形态学像操作的布尔逻辑的简化表达式, 在模拟的实验装置上, 完成了形态学像操作的实验。由于无需光学循环, 因此, 它的结构简单而且易于实现。

关键词 逻辑列阵, 逻辑网络, 数学形态学,

1 引 言

数学形态学是人们从几何测量领域发展起来的一门学科, 该学科逐步形成了一整套以集合论为基础的、严格的、系统的数学理论, 这些理论是图像处理领域中特别有效的数学工具^[1]。利用光学系统来实现数学形态学的各种像操作是人们在这方面的一种尝试。现在已经发展起多种可用光学方法来实现数学形态学像操作的结构体系, 具有代表性的有三种。1) 由Huang等人提出的双值像代数(Binary Image Algebra)体系^[2], 该方法可以用对三种基本操作(Dilation, Union, Complement)的硬件的循环运用来实现所有的数学形态学像操作^[3]; 2) Fukui等人提出的像逻辑代数(Image Logic Algebra)体系^[4], 该方法在双值像代数的基础上, 增加了几种领域像逻辑操作, 从而可以实现包括形态学操作在内的更普遍的像操作; 3) 刘立人等提出的单一算子像代数(One-operation Image Algebra)体系^[5], 该方法在形态学处理单元后接一个逻辑处理单元, 经过循环运用后也可实现所有的形态学像处理功能。这些方法都是从形态学像操作的构成的角度来实现所有的像操作, 并且都需要光学循环, 而在现有技术条件下, 光学循环的实现是比较困难的。

在数学形态学理论中, 像操作是用集合与集合的运算来定义的, 需要用到的运算包括集合的平移和“交”, “并”, “补”运算, 而这些集合运算都可以用列阵逻辑来实现, 例如, 集合间的“交”运算可以用列阵逻辑操作“与”来实现, “并”运算可以用逻辑操作“或”来实现, “补”运算可用列阵逻辑操作“非”来实现, 而集合的平移则用列阵的位移来实现。这样, 人们可以用不同的列阵逻辑操作来表示不同的集合运算形式, 从而实现不同的形态学像操作。本文首先从逻辑操作的角度出发, 提出了一种由逻辑列阵实现的形态学像处理器, 并得到数学形态学

* 高科技办公室和国家自然科学基金委员会资助项目。

收稿日期: 1995年4月26日; 收到修改稿日期: 1995年8月30日

基本像操作的简化以后的布尔逻辑函数,在此基础上,给出了模拟实验装置和实验结果。该装置结构简单而易于实现。

2 逻辑列阵和形态学像处理器

逻辑列阵是指每一个列阵元素都是一个逻辑功能单元的列阵,这里的逻辑列阵中每一单元实现的是同样的逻辑操作。由一般的逻辑理论知道:单一的逻辑列阵不能实现复杂的逻辑函数,要实现复杂的逻辑函数必须采用逻辑网络的形式^[6]。逻辑网络分二级网络和三级网络两种。为简单起见,本文采用二级网络的形式。二级逻辑网络是指信号在输出前要经过两级逻辑门的逻辑网络,其中第一级定义为含有输出逻辑门的那一级,接收输入信号的逻辑门属于第二级。这样的二级逻辑网络共有四种不同的形式,包括与非-与非网络,或非-或非网络,与-或网络,或-与网络。本文用两个逻辑列阵构成一个二级逻辑网络来实现形态学像操作其结构如图 1 所示,它可以实现所有的形态学像处理功能。

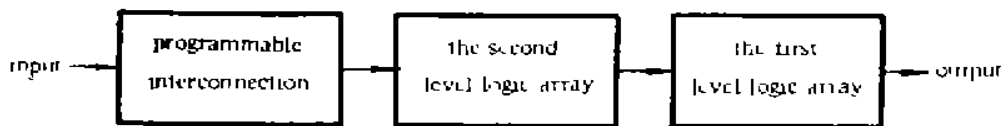


Fig. 1 The architecture of two-stage network by two logic arrays

对于某一个一般的数学形态学像操作来说,依据前面提到的转换原则,很可能得到一个很复杂的逻辑函数。为了使像处理更简单,通常需要简化成标准的逻辑函数形式。根据逻辑理论,所有的综合逻辑函数都可用多种方法来进行简化,(包括文氏图,卡诺图和真值表)而简化的结果一般都表示成“积之和”(与或)或者“和之积”(或与)的标准形式。由于逻辑函数和的形式和逻辑函数积的形式可以很方便地用对偶定理进行转化,因此“积之和”的形式和“和之积”的形式可以很容易地进行转化。

3 基本形态学像操作的简化布尔逻辑函数

用一个列阵 X 来表示一个二维平面双值像, $x(i, j)$ 表示在位置 (i, j) 点的像素的值,它是一个双值变量, $x(i, j) = "0"$ 表示在位置 (i, j) 上的像素是暗点,而 $x(i, j) = "1"$ 表示在位置 (i, j) 上的像素是亮点,这里“0”和“1”是布尔逻辑值。这样,根据数学形态学像操作的定义,可以把所有的形态学像操作作用像素值之间的逻辑操作来表示。

在数学形态学诸操作中膨胀(Dilation)操作和腐蚀(Erosion)操作被认为是两种最基本的像操作,其它的操作可由这两种操作表示。在构成形态学像处理器方面这两种像操作也具有很重要的作用。首先把这两种操作表示成布尔逻辑形式。在形态学中,一个结构核 S 对一个像集 X 的膨胀操作定义如下:结构核 S 的每一个元素都使像集 X 产生一个位移,位移大小由元素的坐标大小决定,膨胀的结果是所有位移像集的并集。该定义可用集合论公式表示如下:

$$Y = X \oplus S = \{y \mid \exists_{s_i \in S} \exists_{x_j \in X} (y = x_j + s_i)\} \quad (1)$$

式中 Y 表示形态学像操作的结果, y 是集合 Y 中的元素。而用布尔逻辑来表示,则膨胀结果可表示如下:

$$Y = \{y(m, n) = \sum_{(i, j) \in S} x(m + i, n + j) \mid (m, n), (m + i, n + j) \in T\} \quad (2)$$

式中 $y(m, n)$ 表示像 Y 在坐标 (m, n) 处的像值, Σ 表示多输入的逻辑或, T 表示像 X 的所有坐标的集合。腐蚀操作要用补集的概念来定义, 一个结构核 S 对一个像集 X 的腐蚀操作定义如下: 结构核 S 的每一个元素都使像集 X 的补集产生一个位移, 位移大小由元素的坐标大小决定, 腐蚀的结果是所有位移像集的并集的补集。同样, 该定义也可用集合论公式表示如下:

$$Y = X \ominus S = \{y \mid \exists_{S_i \in S} \exists_{x_i \in X^c} (y = x_i - S_i)\}^c \quad (3)$$

式中上标 c 表示补集。而用逻辑函数表示, 则具有如下形式:

$$Y = \{y(m, n) = \prod_{(i, j) \in S} x(m - i, n - j) \mid (m, n), (m - i, n - j) \in T\} \quad (4)$$

式中 Π 表示多输入的逻辑与。同样地, 也可以得到形态学的其它一些基本像操作的逻辑表达式。这些操作包括开运算 (Opening)、闭运算 (Closing)、薄化操作 (Thinning)、厚化操作 (Thickening) 和击中与否操作 (Hit-Missing Transform) 等等, 这些操作的简化结果表示如下: 开运算是腐蚀后接着膨胀的结果:

$$Y = X \circ S = (X \ominus S) \oplus S = \{y(m, n) = \sum_{(i, j) \in S} \prod_{(i', j') \in S} x(m + i - i', n + j - j') \mid (m, n), (m + i' - j', n + j' - j) \in T\} \quad (5)$$

闭运算则是膨胀后接着腐蚀的结果:

$$Y = X \bullet S = (X \oplus S) \ominus S = \{y(m, n) = \prod_{(i, j) \in S} \sum_{(i', j') \in S} x(m + i' - i, n + j' - j) \mid (m, n), (m + i' - j', n + j' - j) \in T\} \quad (6)$$

击中与否是一个图像匹配操作:

$$Y = X \otimes S = \{y(m, n) = \prod_{(i_1, j_1) \in S(1)} x(m - i_1, n - j_1) \prod_{(i_2, j_2) \in S(2)} \overline{x(m - i_2, n - j_2)} \mid (m, n), (m - i_1, n - j_1), (m - i_2, n - j_2) \in T\} \quad (7)$$

式中 $S = [S(1), S(2)]$, $S(1)$ 是图像本身匹配的模式, $S(2)$ 则是图像背景匹配的模式。从上式可以看出, 击中与否 (HMT) 变换的逻辑操作包含有像素值和像素值的补, 是一个完整的逻辑“与”的形式, 这是一个重要的逻辑函数形式, 通过改变结构核的形式, 它可以表示所有的逻辑“与”函数项。薄化操作是图像本身与图像的击中与否变换之间的差:

$$Y = X \odot S = X / (X \otimes S) = \{y(m, n) = x(m, n) \left[\sum_{(i_1, j_1) \in S(1)} \overline{x(m - i_1, n - j_1)} + \sum_{(i_2, j_2) \in S(2)} x(m - i_2, n - j_2) \right] \mid (m, n), (m - i_1, n - j_1), (m - i_2, n - j_2) \in T\} \quad (8)$$

厚化操作是图像本身与图像的击中与否变换之间的和:

$$Y = X \oplus S = X \cup (X \otimes S) = \{y(m, n) = x(m, n) + \prod_{(i_1, j_1) \in S(1)} x(m - i_1, n - j_1) + \prod_{(i_2, j_2) \in S(2)} \overline{x(m - i_2, n - j_2)} \mid (m, n), (m - i_1, n - j_1), (m - i_2, n - j_2) \in T\} \quad (9)$$

4 光学实现

从上面的讨论可以知道, 所有的形态学像操作都可以表示成布尔逻辑的形式, 并且所有可能的逻辑函数都能表示成“与或”或者“或与”的形式, 这是两级逻辑网络的形式, 只要用光学系统实现了“与或”和“或与”两级列阵逻辑就可以实现所有的双轨形态学像操作。

本文设计的光学模拟实验装置如图 2 所示。在上节所述的多个逻辑函数中, 包含有输入

变量和它的补变量,因此,在实验装置中,采用双轨编码的形式^[7],即输入平面包含有被操作的像以及它的补像,而结构核则是一个可变的输入。输入平面、结构核以及其后的成像透镜和接收平面构成了一个两通道光学相关器。相关器可以完成输入像素之间的互连。该相关结果被 CCD 采集后,输入到计算机中,并由计算机完成逻辑列阵功能,最后由计算机屏幕输出结果。计算机首先实现的是一个阈值逻辑,它将 CCD 采集的数字化结果通过取阈值的方式完成简单逻辑“与”或者逻辑“或”,得到二值结果后存在计算机中,并和随后每次结构核变化后的结果进行时序逻辑“或”或者逻辑“与”,从而完成列阵逻辑功能。简单的阈值逻辑的输入输出关系如图 3 所示。采用计算机来完成列阵逻辑功能,是因为实验室没有合适的光学阈值和

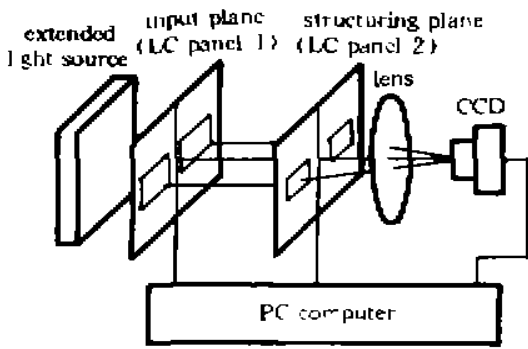


Fig. 2 Construct of the experimental system

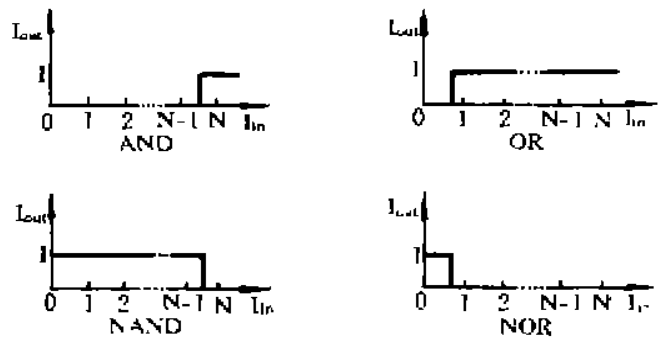


Fig. 3 The format of the N-input thresholding gate

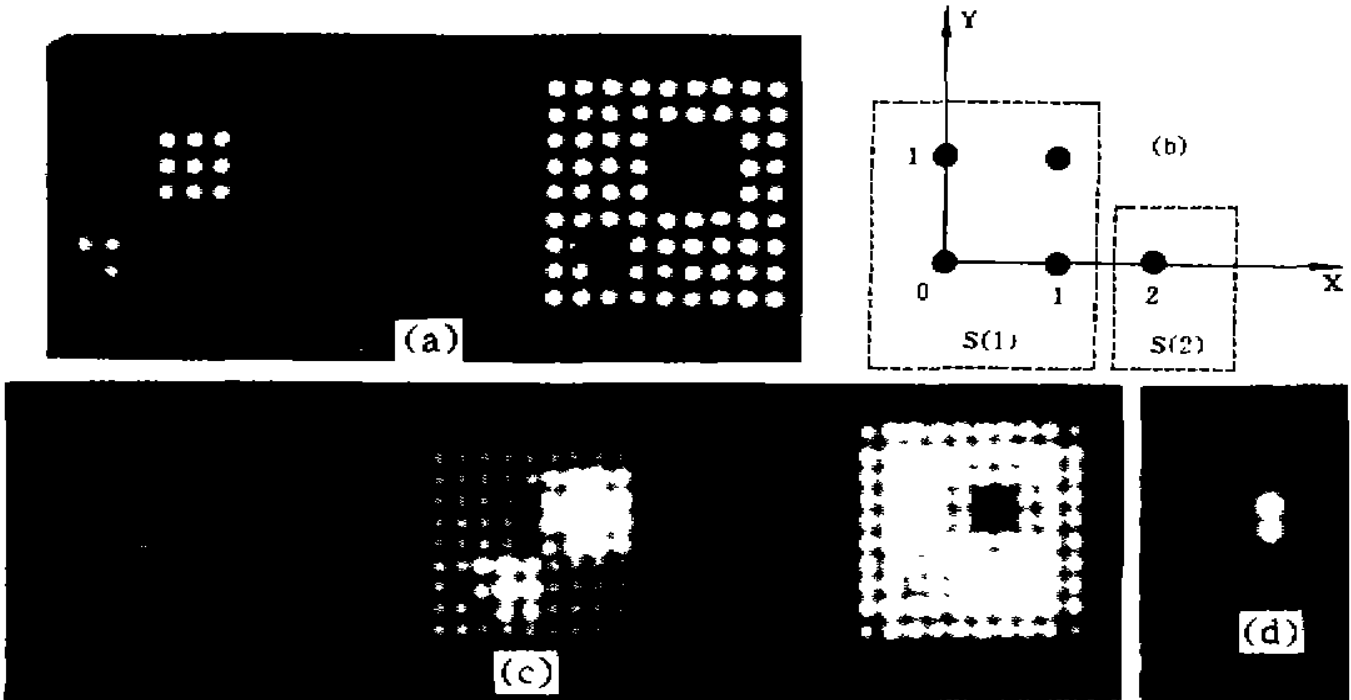


Fig. 4 Experimental result; (a) input image; (b) structuring pair $S = (S(1), S(2))$; (c) the relating result; (d) the result after thresholding

逻辑器件,例如可以对光产生非线性响应的 PROM 器件^[9],以及可以完成光学时序逻辑的 S-SEED 器件^[10]等。S-SEED 器件是由两个完全相同的 SEED 器件串联而成的,其完成的逻辑功能主要是由两个器件上的入射把另外一束偏置光做为另一个器件的入射光。这样,当偏置光功率高时,可以完成阈值逻辑“与”;偏置光功能低时,可以完成阈值逻辑“或”。

在图 2 所示的实验装置上完成了形态学双值像操作中中与否变换的实验。图 4 所示是实验结果。图 4(a)是输入像,图 4(b)是结构核,图 4(c)是相关结果,而图 4(d)则是由计算机

得到的最后结果。虽然完成的只是击中与否变换, 但该变换具有普遍性, 它可以很方便地实现膨胀, 腐蚀等其它像操作, 例如: 在击中与否变换中, 取 $S(1)$ 为膨胀核, $S(2)$ 为空集, 并让阈值逻辑门实现“或”操作, 就能够实现膨胀像操作, 而取 $S(1)$ 为空集, 取 $S(2)$ 为腐蚀核, 并让阈值逻辑门实现“或”操作, 则能够实现腐蚀像操作。

结束语 本文提出了利用逻辑阵列来实现数学形态学的双值像操作的设想, 并得到了形态学基本像操作的布尔逻辑函数的简化表达式, 在此基础上, 给出了实验装置和实验结果。这是一种新的实现数学形态学像操作的方法, 它适于用光学逻辑器件实现。由于无需光学循环, 其结构很简单而且易于实现, 并且可以得到推广。例如用模糊逻辑代替布尔逻辑, 该结构将可能应用到形态学的灰度像处理上。

参 考 文 献

- [1] P. Maragos, Tutorial on advances in morphological image processing and analysis. *Opt. Engng.*, 1987, 26(7): 623~632
- [2] K.-S. Huang, B. K. Jenkins, A. A. Sawchuk, Binary image algebra and optical cellular logic processor design. *Comput. Vision. Graphics Image Process.*, 1989, 45(3): 295~345
- [3] K.-S. Huang, A. A. Sawchuk, B. K. Jenkins *et al.*, Digital optical cellular image processor (DOCIP): experimental implementation. *Appl. Opt.*, 1993, 32(2): 166~173
- [4] M. Fukui, K. Kitayama Image logic algebra and its optical implementations. *Appl. Opt.*, 1992, 31(5): 581~591
- [5] L. Liu, Z. Zhang, X. Zhuang, One-operation image algebra and optoelectronic cellular two-layer logic array. *J. Opt. Soc. Am. (A)*, 1994, 11(6): 1789~1797
- [6] H. T. Nagle, Jr., B. D. Carroll, J. D. Irwin, *Introduction to Computer Logic*, Prentice-Hall, Inc, 1975
- [7] L. Liu, X. Liu, Cascadable binary pattern logic processor using multiple-imaging. *Opt. Commun.*, 1991, 82(5~6): 446~452
- [8] L. Liu, Optoelectronic implementation of mathematical morphology. *Opt. Lett.*, 1989, 14(10): 482~484
- [9] B. A. Horwitz, F. J. Corbett, The PROM-theory and applications for the Pockels readout optical modulator. *Opt. Engng.*, 1978, 17(4): 353~364
- [10] A. L. Lentine, H. S. Hinton, D. A. B. Miller *et al.*, Symmetric self-electrooptic effect devices; optical set-reset latch, differential logic gate, and differential modulator/detector. *IEEE J. Quant. Electron.*, 1989, QE-25(8): 1928~1936

Mathematical Morphological Binary Image Processor Using Logical Array

Wang Binqun Liu Liren Liang Feng

(Information Optics Laboratory, Shanghai Institute of Optics and Fine Mechanics,
Chinese Academy of Sciences, Shanghai 201800)

(Received 26 April 1995; revised 30 August 1995)

Abstract From the point of view of logic operation implementation of mathematics morphological binary image operations using logical array is proposed in this paper, and some basic morphological image operations' minimized logic representations are given. The experiments of morphological image operation have been demonstrated in the simulated architecture. Since the feedback loops are not necessary, so it is simple and easy to execute.

Key words logic array, logic network, morphology, correlator.