

# 基于围线扫描的快速完全欧氏距离变换算法

鲁继文 张二虎

(西安理工大学信息科学系, 西安 710048)

**摘要** 欧几里德距离变换(EDT)是由黑白像素构成的二值图像中所有像素找到距离其最近的黑色像素,并计算它们之间的欧几里德距离,目前广泛地应用于图像分析和计算机视觉等领域。本文采用基于围线扫描的思想,提出了一个在二值图像中进行完全欧氏距离变换的算法。算法首先将二值图像中的像素进行分类,对那些本身既不是特征像素且其4-邻域内也没有特征像素的点作上标记,然后对这些标记的像素自内向外进行围线扫描,搜索与它最近的黑点并计算它们的欧氏距离。算法能够计算精确的欧氏距离。同时对算法的时间复杂度进行了简单的分析,并给出了程序实现中得到的一些实验数据,结果表明该算法运算速度快,时空需求在当前的硬件环境下令人满意,是一种有效的和有着巨大实际应用价值的距离变换算法。

**关键词** 二值图像 围线扫描 完全欧氏距离变换 时间复杂度 图像分析

中图分类号: TP391 文献标识码: A 文章编号: 1006-8961(2006)06-0872-05

## A Fast Algorithm for Complete Euclidean Distance Transformation Based on Contour Scanning

LU Ji-wen, ZHANG Er-hu

(Department of Information Science, Xi'an University of Technology, Xi'an 710048)

**Abstract** The Euclidean Distance Transform(EDT) converts a binary image into one where each pixel has a value equal to its Euclidean distance to the nearest foreground. It has important applications in image analysis and computer vision. This paper gives an algorithm for complete Euclidean distance transformation on the binary image on the basis of the contour scanning. In the suggested algorithm all the pixels are classified and the ones which neither are characteristic pixel nor have characteristic pixels in 4-neighbors are marked. For the marked pixel, scan its contour from inner to outer, find its nearest characteristic pixel and compute their Euclidean distance. The exact distances are obtained. The requirement of time are analyzed briefly, some testing data produced in running the implementation are illustrated. The result shows that the algorithm runs fast, the requirements of space and time are satisfying in the present hardware condition, and it is an effective and enormous applied distance transformation algorithm.

**Keywords** binary image, contour scanning, complete Euclidean distance transform, time complexity, image analysis

## 1 引言

自从 Rosenfeld 和 Pfaltz<sup>[1]</sup>于 1966 年首次提出距离变换概念以来,距离变换被广泛应用于图像分析、模式识别与计算机视觉等领域,并已逐渐成为其中的一个基本运算和研究热点,人们利用它

实现目标细化、骨架抽取、图像合并、分割与配准<sup>[2-4]</sup>等,尤其是在医学图像配准中取得了巨大的成功,因此其实现算法受到人们极大的重视。在 2 维空间,一个  $n \times n$  的二值图像可以认为仅包含特征和背景两种像素。将特征像素称为“黑点”,背景像素称为“白点”,距离变换就是为二值图像中的每一个像素寻找离它最近的黑点像素,

收稿日期:2004-08-25;改回日期:2005-08-29

第一作者简介:鲁继文(1981~),男,助教。现为西安理工大学信号与信息处理专业硕士研究生。主要研究方向为数字图像处理与模式识别等。E-mail:lujiwen@xaut.edu.cn.

并计算它们之间的距离,同时在距离变换图像中的每个像素值存在该像素到离它最近的一个黑点的距离。在实际计算中,常采用两种距离测度:非欧几里德距离和欧几里德距离(Euclidean distance)。前者常采用的是城市街区、斜切等距离,它们虽然计算简单,但仅是对欧几里德距离变换(EDT)的一种近似,在许多对精度要求较高的计算中,必须使用 EDT。然而,EDT 计算具有较高的复杂度,在最坏的情形下,其计算量达到  $O(n^4)$ 。近年来出现了许多计算 EDT 的近似算法,如文献[5]利用二值图像中像素的空间相关性,提出了一种统一化的串行距离变换算法,算法运算速度虽然很快,但是其时间复杂度严重依赖于特征点的空间分布,因此对某些像素的计算结果仍然是不准确的,这些算法也常被称为不完全 EDT 算法。为了计算完全 EDT,文献[6]、[7]分别提出了一种复杂度为  $O(n^3)$  和  $O(n^2 \log n)$  的算法,然后进行并行计算。文献[8]提出了一种完全欧氏距离变换的最优算法,采用行列交叉处理的方法,缩小搜索最近黑点的范围,算法复杂度为  $O(n^2)$ 。文献[9]提出了一种基于边过程的围线跟踪的完全 EDT。文献[10]在文献[8]的基础上提出了一种复杂度为  $O(n^3 \log n)$  的 3 维完全欧氏距离变换算法。这些算法都不同程度地存在着计算耗时过多的缺点,基于此,本文采用围线扫描的方法,逐步搜索最近的黑点,最后计算完全 EDT。实验结果表明,本文所提的算法实现简单、计算复杂度较低,优于文献[8]的最优复杂度,不失为一种有效的完全 EDT 算法。

## 2 基本概念及基本思想

一个  $n \times n$  的二值图像可以表示成为 2 维数组  $A = [a_{ij}]$ ,若像素  $(i, j)$  为“黑点”,则  $a_{ij} = 1$ , 否则  $a_{ij} = 0$ 。记  $F = \{(x, y) \mid a_{xy} = 1\}$  为所有黑点的集合,距离变换就是要对二值图像中的所有像素  $(i, j)$ , 计算

$$d_{ij} = \min_{(x,y) \in F} D[(i, j), (x, y)] \quad (1)$$

这里,  $D[(i, j), (x, y)]$  表示  $(i, j)$  到  $(x, y)$  的距离。

下面以二值图像  $a_{ij}$  中的任一像素  $P_i$  为例,完全欧氏距离变换的任务就是寻找离  $P_i$  最近的黑点并计算它们之间的距离

$$D[(i, j), (x, y)] = \sqrt{(i-x)^2 + (j-y)^2} \quad (2)$$

如果  $P_i$  为黑点,则离它最近的黑点就是它本身,此时它们之间的欧几里德距离为 0;如果  $P_i$  不是黑点,则看它的 4-邻域内的 4 个像素是否有一个像素为黑点,若它们中至少有一个像素为黑点,则  $P_i$  到该黑点的欧几里德距离为 1;如果 4-邻域内的 4 个像素也没有一个像素为黑点,则开始对  $P_i$  像素周围的一个围线区域进行扫描搜索,围线区域由以  $P_i$  为中心的一个  $(2r+1) \times (2r+1)$  ( $r=1, 2, \dots, n-1$ ) 的正方形区域组成,  $r=1$  时围线区域如图 1 所示(这里  $r$  表示围线区域中心像素到围线的距离)。

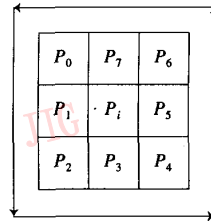


图 1 扫描以  $P_i$  为中心的  $3 \times 3$  围线区域

Fig.1 Scanning the  $3 \times 3$  contour area centered with  $P_i$

当  $r=1$  时,围线区域即为  $P_i$  像素的 8-邻域,搜索按照  $P_0, P_1, \dots, P_7, P_0$  逆时针方向的顺序进行。若  $r=1$  时,在  $(2r+1) \times (2r+1)$  的围线区域内没有找到黑点,此时需要将搜索区域的正方形边长  $(2r+1)$  扩大,取  $r=2$ ,在一个  $5 \times 5$  围线区域进行扫描搜索,搜索顺序如图 2 所示,从  $P_0$  开始沿箭头方向,依次搜索  $P_0, P_1, \dots, P_{15}$  逆时针方向直至回到  $P_0$ 。

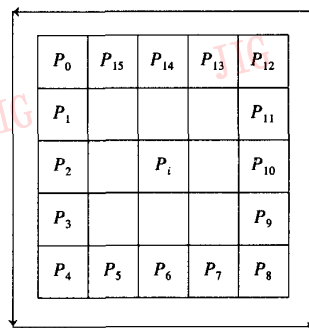


图 2 扫描以  $P_i$  为中心的  $5 \times 5$  围线区域

Fig.2 Scanning the  $5 \times 5$  contour area centered with  $P_i$

同理,当扫描完以  $P_i$  为中心的  $5 \times 5$  围线区域后,仍然没有找到黑点,通过不断增加  $r$  的值继续扩大扫描的围线区域,直到找到离它最近的黑点,并计算它们之间的欧几里德距离。

### 3 算 法

设输入是一个  $n \times n$  的二值图像矩阵  $b[i][j]$ ,  $1 \leq i, j \leq n$ , 输出的距离变换结果存放在 2 维数组  $R[n][n]$  中。

(1) 标记 逐行扫描二值图像  $b[i][j]$ , 若扫描的当前像素值为 1, 将此类像素记为 A 类像素; 若扫描的当前像素值为 0, 但其 4-邻域内的 4 个像素中至少有一个像素为 1, 将此类像素记为 B 类像素; 若扫描的当前像素值不为 1, 且其 4-邻域的 4 个像素中也没有一个像素为 1, 则将此像素记为 C 类像素;

①若扫描的当前像素为 A 类像素, 则在距离变换结果数组  $R$  中相应的位置处置为 0;

②若扫描的当前像素为 B 类像素, 则在距离变换结果数组  $R$  中相应的位置处置为 1;

③若扫描的当前像素为 C 类像素, 则在距离变换结果数组  $R$  中相应的位置处存储其前一个 C 类像素的位置, 构成一个后向链表, 其中扫描到第 1 个 C 类像素为表的最末一个元素, 扫描到最后一个 C 类像素为表的第 1 个元素。这样, 在计算完一个 C 类像素的距离后, 根据其存储的前一个 C 类像素的位置, 就可以跳转到前一个 C 类像素处, 而节省了搜索的时间。

(2) 对第 1 步中的情况 ③ 进行围线扫描搜索, 计算各 C 类像素的距离。

①从链表的第 1 个元素开始, 对于 C 类的每一个像素, 分别对其围线区域进行扫描搜索, 初始  $r = 1$ , 围线区域由一个  $(2r + 1) \times (2r + 1)$  ( $r = 1, 2, \dots, n - 1$ ) 的正方形环组成, 正方形环的宽度为单像素宽, 并且  $r = k$  的正方形环围线刚好被  $r = k + 1$  的正方形环围线所包含 ( $k = 1, 2, \dots, n - 1$ )。如果当  $r = k$  时找到离当前像素最近的点, 计算它们之间的距离  $D_{old}$ , 然后每次将  $r$  不断增 1, 继续搜索新的  $(2r + 1) \times (2r + 1)$  围线区域, 在新的围线区域上寻找离中心像素最近的点, 并计算它们之间的距离  $D_{new}$ , 每次将  $D_{new}$  与  $D_{old}$  比较, 取  $D = \min\{D_{old}, D_{new}\}$ , 直到  $D \geq r$  时, 记录此时的  $D$  值, 距离当前像素最近的点的欧氏距离为  $D$ , 存放在距离变换结果数组  $R$  对应的位置处。

②根据当前 C 类像素处存放的前一个 C 类像素的位置, 跳转到前一个像素处, 返回到步骤 ① 继续进行处理, 直到最末一个 C 类像素处理完。

(3) 最后将距离变换结果数组  $R$  输出, 算法结束。

### 4 方法评述与实验结果

容易看出算法的正确性: 算法中第 2 步是最花费时间的, 同样它也保证了为 C 类每个像素找到离它最近的点, 它们之间的欧几里德距离为最优, 因为它每次总是将与当前像素欧几里的距离最小的像素点保留下来, 所以能够保证获得的欧几里德距离值是正确的也是最优的。

下面讨论算法的计算时间。假设扫描并处理一个像素所需的时间为  $t_s$ , 计算两个像素之间的距离  $t_w$ , 由文献 [8] 证明知, 完全欧几里德距离变换的最优算法的时间为  $4n^2 t_s + (22n^2 - 18n)t_w$ , 由于  $t_w$  远大于  $t_s$ , 在算法中距离计算占据了大量的计算时间, 因此算法总的计算时间为  $(22n^2 - 18n)t_w$ , 证明详见文献 [8]。

对于一幅  $n \times n$  的二值图像, 一般地, 其特征点越少, 计算所消耗的时间就越多。显然, 当二值图像中只有一个特征点, 且位于图像的边角处 (图像左上角、右上角、左下角、右下角中任何一角处), 此时所需的计算时间是最大的, 所耗费的计算成本最高, 因为其他的所有非特征点都必须且只有搜索到该特征点才能找到离它最近的特征点, 如果图像中还有其他的特征点, 则有些非特征点就不用搜索到图像边角处的特征点就找到了离它最近的特征点, 也不会出现最大的欧氏距离  $\sqrt{2}(n - 1)$ , 可以证明算法此时的计算时间将小于  $n^2 t_w + 8n^4 t_s$ 。

假设图像中唯一一个特征点位于原二值图像的左上角, 此时应该是为图像中右下角那个像素寻找离它最近的特征点所需花费的时间最多, 要搜索到左上角的那个特征点, 它需要搜索一个  $(2\sqrt{2}(n - 1) + 1) \times (2\sqrt{2}(n - 1) + 1)$  的正方形环围线区域, 计算一次两点间距离, 总共所需的时间为  $t_w + (2\sqrt{2}(n - 1) + 1)^2 t_s$ , 对于一幅  $n \times n$  的二值图像, 其他像素点的计算时间肯定小于  $t_w + (2\sqrt{2}(n - 1) + 1)^2 t_s$ , 为讨论简便起见, 将其计算时间均设定为该时间, 这样花费的总时间  $T$  将小于  $n^2 t_w + n^2 (2\sqrt{2}(n - 1) + 1)^2 t_s$ , 即  $T < n^2 t_w + 8n^4 t_s$ , 故当

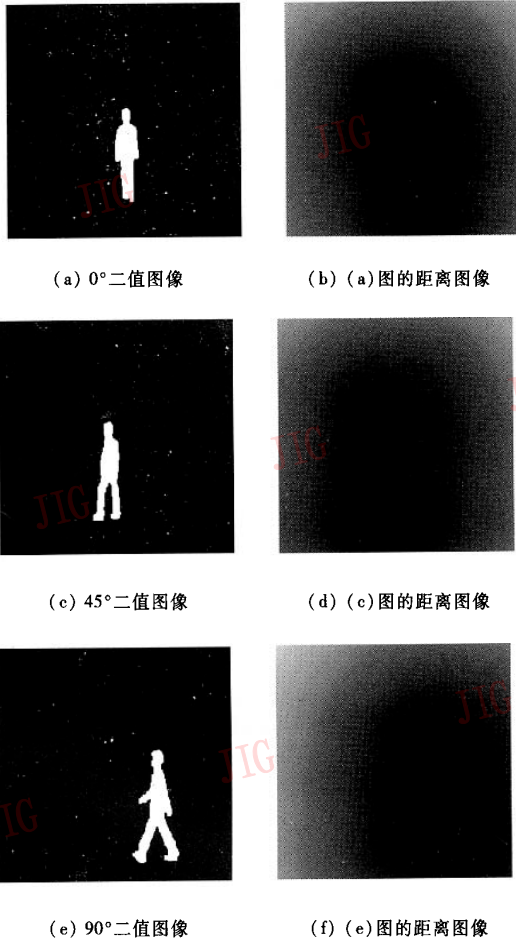
$$\frac{t_w}{t_s} \geq \frac{8n^3 - 4n}{21n - 18} \quad (3)$$

有

$$(n^2 t_w + 8n^4 t_s) < 4n^2 t_s + (22n^2 - 18n)t_w \quad (4)$$

一般的,  $t_w$  远大于  $t_s$ , 因此本文算法与文献[8]算法相比, 计算时间显然优于文献[8]算法。

图3是以视频序列中某一帧的人体运动轮廓为对象, 采用本文提出的快速欧几里德距离变换执行EDT运算, 可以很方便地得到欧几里德距离图像。



(a) 0°二值图像

(b) (a)图的距离图像

(c) 45°二值图像

(d) (c)图的距离图像

(e) 90°二值图像

(f) (e)图的距离图像

图3 EDT运算前后的二值图像与距离图像

Fig.3 Original binary images and the distance images after EDT

其中, (a), (c)和(e)分别对应着人的运动方向与摄像机成0°, 45°和90°时的人体轮廓, (b), (d)和(f)分别为对应的欧几里德距离图像。通过获得的在不同视角下的欧几里德距离图像, 可以实现人体运动过程中的多摄像机跟踪与融合。

在赛场1.7G微机对上对16×16至512×512的选取的或随机的许多二值图像进行了实验, 表1中仅列出了其中6个例子的实验结果。从表1的数据对比可以看出, 本文算法优于文献[8]的最优算法, 但随着二值图像大小的增大, 运算时间提高的倍数

会有所降低, 当图像大小由128×128增大为512×512时, 本文算法较文献[8]最优算法的运算速度由20倍减至10倍, 即原图像越大, 本算法的优势会越弱。但是一般在实际应用中, 图像的大小都不是很大, 因此本文算法的优越性能得以较好地体现。

表1 实验运行的6个例子

Tab.1 Six examples of our experimental results

	二值图像		文献[8]算法 运行时间(s)	本文算法 运行时间(s)
	大小	特征像素个数		
例1	16×16	1	可忽略	可忽略
例2	32×32	94	1	可忽略
例3	64×64	401	5	可忽略
例4	128×128	1692	20	1
例5	256×256	6457	50	3
例6	512×512	25519	250	25

## 5 结论

本文提出了一种2维完全欧氏距离变换的新方法。通过理论分析及实验可以看出, 其运算时间优于文献[8]的最优算法运算时间, 同时本文算法计算的是精确的欧氏距离, 算法思路直观, 编程实现方便, 在进行2维图像的完全欧几里德距离变换时取得了良好的效果。大量实验结果表明, 算法的运算速度较快, 占用的内存较少, 时空需求在当前的硬件环境下可以令人满意, 因此是一种很有前途的距离变换算法, 有着巨大的实际应用价值。当然, 如何更好地将本算法嵌入到一些硬件平台上, 如DSP, FPGA等是今后进一步的研究重点。

## 参考文献 (References)

- Rosenfeld A, Pfaltz J L. Sequential operations in digital picture processing[J]. Journal of the ACM, 1966, 13(4):471~494.
- Rosenfeld A, Pfaltz J. Distance functions on digital picture[J]. Pattern Recognition, 1968, 1(1):33~61.
- Fischler M A, Barrett P. An iconic transform for sketch completion and shape abstraction[J]. Computer Graphics and Image Processing, 1980, 13(3):334~360.
- Ye Q Z. The signed Euclidean distance transform and its application [A]. In: Proceeding of Ninth International Conference on Pattern Recognition[C], Rome, Italy, 1988:495~499.
- Guan Wei-guang, Ma Song-de. A fast unified distance transformation algorithm[J]. Chinese Journal of Computers, 1995, 18(8):

- 626 ~ 635. [管伟光, 马颂德. 统一化的快速距离变换[J]. 计算机学报, 1995, 18(8):626 ~ 635.]
- 6 Yamada H. Complete Euclidean distance transform by parallel operation[A]. In: Proceedings of Seventh International Conference on Pattern Recognition [C], Montreal, Quebec, Canada, 1984: 69 ~ 71.
- 7 Kolountzakis M N, Kutulakos K N. Fast computation of the Euclidean distance maps for binary images[J]. Information Processing Letters, 1992, 43(3):181 ~ 184.
- 8 Chen Ling. Optimal algorithm for complete Euclidean distance transform [J]. Chinese Journal of Computers, 1995, 18(8): 611 ~ 616. [陈凌. 完全欧几里德距离变换的最优算法[J]. 计算机学报, 1995, 18(8):611 ~ 616.]
- 9 Wang Zheng-xuan, Li Wen-hui, Pang Yun-jie. An algorithm for complete Euclidean distance transformation based on contour tracing [J]. Chinese Journal of Computers, 1998, 21(3):217 ~ 222. [王征旋, 李云辉, 庞云阶. 基于围线追踪的完全欧氏距离变换算法[J]. 计算机学报, 1998, 21(3):217 ~ 222.]
- 10 Zhu-ge Ying, Tian Jie, Wang Wei-hong. A new method of three-dimensional Euclidean distance transformation [J]. Journal of Software, 2001, 12(3):383 ~ 389. [诸葛婴, 田捷, 王蔚洪. 三维欧氏距离变换的一种新方法[J]. 软件学报, 2001, 12(3): 383 ~ 389.]