

## 三维欧氏距离变换的一种新方法\*

诸葛婴<sup>1</sup>, 田捷<sup>1</sup>, 王蔚洪<sup>2</sup>

<sup>1</sup>(中国科学院自动化研究所人工智能实验室, 北京 100080);

<sup>2</sup>(北京理工大学电子工程系, 北京 100081)

E-mail: tian@doctor.com

http://www.3dmed.net

**摘要:**常见的三维距离变换算法大都是对城市街区、棋盘等二维近似欧氏距离变换算法的三维扩展,得到的依然是近似欧氏距离.提出一种新的三维欧氏距离变换算法,可以得到完全欧氏距离,时间复杂度为 $O(n^3 \log n)$ .将该算法应用于三维医学CT图像内部软组织的显示,取得了较好的效果.

**关键词:**距离变换;复杂度;软组织显示

**中图法分类号:** TP391 **文献标识码:** A

自从 Rosenfield 和 Pfalt<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与模式识别领域.在二维空间,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.我们将特征像素称为“黑点”,背景像素称为“白点”.在距离变换图像中,每个像素值表示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:非欧氏距离和欧氏距离.前者常用的是城市街区(city block)、切削(chamfer)等距离.算法采用串行扫描实现距离变换,在扫描过程中传递最短距离信息,计算简单,但得到的仅是欧氏距离的一种近似值.文献[2]提出一种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,算法复杂度达到 $O(n^2)$ .

常见的三维距离变换算法大都是对二维近似欧氏距离变换算法的三维扩展,Okabe<sup>[3]</sup>等人将城市街区/棋盘方法扩展到三维. Borgfors<sup>[4]</sup>设计了一个 $3 \times 3 \times 3$ 的局部操作器来实现距离变换,并将算法产生的距离与欧氏距离的差值最小作为优化标准.这些算法得到的都是近似的欧氏距离.而关于三维欧氏距离变换算法,文献中却很少提及.1993年,Ragnemalm<sup>[5]</sup>对此算法做了改进,但仍然存在着一定的误差.在有些应用领域,如三维医学图像处理中,由于对精度有很高的要求,因此要求距离变换算法得到的必须是完全欧氏距离.

本文在文献[2]提出的二维欧氏距离变换算法的基础上,提出了一种三维欧氏距离变换算法,在三维空间中可以得到完全欧氏距离.对一个 $n \times n \times n$ 的三维二值图像,我们将其分解成 $n$ 个 $n \times n$ 的二维二值图像,首先对 $n$ 个二值图像进行二维欧氏距离变换,然后利用二维欧氏距离变换的结果,将每个二维图像中的像素与其他二维图像中的黑点进行距离比较,通过优化方法来减少需要参与距离计算、比较的黑点个数.整个算法的时间复杂度为 $O(n^3 \log n)$ .

在医学图像处理中,常用直接体绘制(volume rendering)技术进行三维体数据绘制,医学 CT

• 收稿日期: 1999-05-04; 修改日期: 1999-12-27

基金项目: 国家自然科学基金资助项目(69843001), 国家 863 高科技发展计划资助项目(863-306-ZD04-06-4)

作者简介: 诸葛婴(1971-),男,浙江人,博士,主要研究领域为计算机图像处理,模式识别;田捷(1960-),男,安徽人,研究员,博士生导师,主要研究领域为多媒体技术,计算机网络;王蔚洪(1974-),女,北京人,硕士,主要研究领域为计算机图像处理,模式识别.

图像内部的肌肉、神经血管等部分由于受表皮及皮下脂肪的影响而无法在最终的显示图像中体现出来. 本文利用三维欧氏距离变换算法, 首先将医学 CT 图像中外层的表皮及皮下脂肪剥去, 然后进行体绘制, 使内部肌肉、神经血管可以很好地显示出来.

本文第 1 节介绍文献[2]提出的二维欧氏距离变换算法. 第 2 节描述三维欧氏距离变换算法. 在第 3 节给出三维欧氏距离变换算法在三维医学 CT 图像内部软组织显示中的应用. 最后在第 4 节对全文做小结.

## 1 二维欧氏距离变换算法

一个  $n \times n$  (本文假定  $n = 2^m$ ,  $m$  为正整数) 的二值图像用二维数组  $A = [a_{ij}]$  来表示, 若像素  $(i, j)$  为黑点, 则  $a_{ij} = 1$ , 否则  $a_{ij} = 0$ . 记  $M(i, j)$  为距离像素  $(i, j)$  最近的黑点,  $M(i, j)$  到  $(i, j)$  的欧氏距离即为像素  $(i, j)$  进行欧氏距离变换的结果. 算法的目的就是尽量缩短求  $M(i, j)$  的时间.

文献[2]中有如下一些定义:

**定义 1.** 记  $F = \{(i, j) | a_{ij} = 1\}$  为图像中所有黑点的集合.

**定义 2.** 对于给定  $(i, j)$ , 记

$SW_{ij} = F \cap \{(x, y) | x \leq i, y \leq j\}$  为  $(i, j)$  左下部分所有黑点的集合;

$SE_{ij} = F \cap \{(x, y) | x \leq i, y > j\}$  为  $(i, j)$  右下部分所有黑点的集合;

$NW_{ij} = F \cap \{(x, y) | x > i, y \leq j\}$  为  $(i, j)$  左上部分所有黑点的集合;

$NE_{ij} = F \cap \{(x, y) | x > i, y > j\}$  为  $(i, j)$  右上部分所有黑点的集合.

这 4 个集合并到  $(i, j)$  最近的黑点分别记为  $SW(i, j)$ ,  $SE(i, j)$ ,  $NW(i, j)$  与  $NE(i, j)$ , 它们当中距离  $(i, j)$  最近者即为  $M(i, j)$ .

**定义 3.**  $L(i, j) = \max\{l; (i, j) \in W_l\}$ , 其中  $W_l$  表示第  $j$  列像素左边所有黑点的集合, 则  $(i, L(i, j))$  为  $W_l$  中第  $i$  行最右边的黑点;  $R(i, j) = \min\{l; (i, l) \in E_j\}$ , 其中  $E_j$  表示第  $j$  列像素右边所有黑点的集合, 则  $(i, R(i, j))$  为  $E_j$  中第  $i$  行最左边的黑点;  $P(i, j) = \max\{l; (l, j) \in S_i\}$ , 其中  $S_i$  表示第  $i$  行像素下边所有黑点的集合, 则  $(P(i, j), j)$  为  $S_i$  中第  $j$  列最上边的黑点;  $Q(i, j) = \min\{l; (l, j) \in N_i\}$ , 其中  $N_i$  表示第  $i$  行像素上边所有黑点的集合, 则  $(Q(i, j), j)$  为  $N_i$  中第  $j$  列最下边的黑点.

有了这 4 个数组, 可简化对  $SW(i, j)$ ,  $SE(i, j)$ ,  $NW(i, j)$  与  $NE(i, j)$  的寻找. 以  $SW(i, j)$  为例, 不必计算与比较  $(i, j)$  到  $SW_{ij}$  中所有黑点的距离, 而只需将  $(i, j)$  与  $(1, L(1, j))$ ,  $(2, L(2, j))$ ,  $\dots$ ,  $(i, L(i, j))$  作计算比较即可.

**引理 1.** 设  $(a, j)$ ,  $(b, j)$  为同一列上的两个像素, 且  $b < a$ , 若  $SW(a, j) = (x, y)$ ,  $SW(b, j) = (z, w)$ , 则  $x \leq z$ ; 设  $(i, a)$ ,  $(i, b)$  为同一行上的两个像素, 且  $b < a$ , 若  $SW(i, a) = (s, t)$ ,  $SW(i, b) = (u, v)$ , 则  $v \leq t$ .

证明详见文献[2]. 对  $SE$ ,  $NW$  与  $NE$  也有同样的结论.

由引理 1, 可进一步缩小对最近黑点的搜索范围, 如已知  $SW(i, j) = (z, w)$ , 则对  $j$  列上的其他像素  $(k, j)$ , 若  $k < i$ , 则  $SW(k, j)$  只能在第  $1 \sim z$  行的范围内; 若  $k > i$ , 则  $SW(k, j)$  只能在第  $z \sim n$  行的范围内. 推而广之, 设第  $j$  列像素已被  $r$  等分且已知  $SW\left(\frac{kn}{r}, j\right) = (z_w, w_k) (k=1, \dots, r)$ , 则数组  $L$  第  $j$  列也被分成  $r$  部分, 其中第  $k$  部分为  $L(x_{k-1}, j) \sim L(x_k, j)$ . 因此, 第  $k$  部分像素只需用数组  $L$  第  $j$  列的第  $k$  部分元素去寻找最近的黑点. 同样地, 如果一行像素被  $r$  等分, 也可采用同样的方法

缩小对  $SW$  值的搜索范围. 因此, 对  $SW$  值的计算可以在行、列方向上交叉进行, 以便互相等分, 减少计算量. 这种行列交叉方式同样适用于对  $SE, NW$  与  $NE$  值的计算.

二维完全欧氏距离变换算法如下:

第 1 步: 扫描整个图像, 计算数组  $L, R, P, Q$ ;

第 2 步: 对所有像素采用行列交叉法计算其  $SW, SE, NW$  与  $NE$  值, 以对  $SW$  值的计算为例来说明行列交叉法的过程. 即先计算第  $\frac{n}{2}$  行上像素的  $SW$  值, 然后再计算第  $\frac{n}{2}$  列上像素的  $SW$  值, 此时, 该列上的像素已被二等分,

对它们计算  $SW$  值所需的计算量大大减少; 再计算  $\frac{n}{4}$  行和  $\frac{3n}{4}$  行, 接着计算第  $\frac{n}{4}$  列和第  $\frac{3n}{4}$  列... 等分越来越细, 则计算量越来越小, 直至所有行、列像素的  $SW$  值全部计算出来为止, 对  $SE, NW$  与  $NE$  值的计算可类似进行;

第 3 步: 对所有像素  $(i, j)$ , 求出  $SW(i, j), SE(i, j), NW(i, j)$  及  $NE(i, j)$  中距离  $(i, j)$  最近者即为  $M(i, j)$ .

整个算法的复杂度为  $O(n^2)$ . 证明详见文献[2].

## 2 三维完全欧氏距离变换

### 2.1 基本思想

一个  $n \times n \times n$  的三维二值图像用三维数组  $A = [a_{ijk}]$  来表示, 若像素  $(i, j, k)$  为特征点, 则  $a_{ijk} = 1$ , 我们称其为“黑点”, 否则  $a_{ijk} = 0$ , 我们称其为“白点”. 记  $V(i, j, k)$  为距离像素  $(i, j, k)$  最近的黑点, 算法的目的是对每一个像素  $(i, j, k)$  以尽可能小的时间复杂度找到它的  $V(i, j, k)$ , 并计算两个像素之间的欧氏距离.

对一个  $n \times n \times n$  的三维二值图像求每一个像素的最近黑点, 最简单的方法就是将每一个白点与三维图像中的每一个黑点进行距离计算, 其中距离最短者即为其最近黑点, 整个算法的复杂度为  $O(n^6)$ . 我们将一个  $n \times n \times n$  的三维二值图像分解成  $n$  个  $n \times n$  的二维二值图像, 首先采用第 1 节所述方法对  $n$  个二值图像进行二维欧氏距离变换, 得到每个像素在其所在二维图像中的最近黑点. 然后利用二维欧氏距离变换的结果, 对每个二维图像中的像素与其他二维图像中的黑点进行距离比较, 通过优化方法来减少需要参与距离计算和比较的二维图像个数与二维图像中黑点的个数.

**命题 1.** 设  $(i, j, k_1)$  与  $(i, j, k_2)$  为两个二维图像上具有相同行列位置的两个像素, 且已知  $(i, j, k_2)$  在其二维图像中的最近黑点为  $(a, b, k_2)$ , 则  $(i, j, k_1)$  在  $(i, j, k_2)$  所在二维图像中的最近黑点亦为  $(a, b, k_2)$ .

证明: 对  $(i, j, k_2)$  所在二维图像中的任意一个像素  $(z, w, k_2)$ , 根据已知条件可知:  $(z - i)^2 + (w - j)^2 \geq (a - i)^2 + (b - j)^2$ , 则

$$(z - i)^2 + (w - j)^2 + (k_2 - k_1)^2 \geq (a - i)^2 + (b - j)^2 + (k_2 - k_1)^2,$$

即  $(i, j, k_1)$  在  $(i, j, k_2)$  所在二维图像中的最近黑点为  $(a, b, k_2)$ . □

由命题 1 可知, 为找到一个像素  $(i, j, k)$  在另一个二维图像中的最近黑点, 不必与该图像中所有黑点进行距离计算与比较, 一次便可命中. 为找到一个像素在三维空间中的最近黑点, 本来需要将其与  $n$  个二维图像中的所有黑点作距离计算与比较, 但应用命题 1, 只需对  $n$  个二维图像中的每一个作一次距离计算, 并作比较即可求得最近黑点. 这大大缩小了在三维空间中对最近黑点的搜索范围.

为减少参与距离计算和比较的二维图像的个数, 我们给出命题 2.

**命题 2.** 设  $(i, j, k_1)$  与  $(i, j, k_2)$  为两个二维图像上具有相同行列位置的两个像素且  $k_1 < k_2$ , 若  $V(i, j, k_1) = (a, b, c), V(i, j, k_2) = (x, y, z)$ , 则  $c \leq z$ .

证明: 根据已知条件  $V(i, j, k_1) = (a, b, c), V(i, j, k_2) = (x, y, z)$ , 则

$$(x-i)^2 + (y-j)^2 + (z-k_1)^2 \geq (a-i)^2 + (b-j)^2 + (c-k_1)^2,$$

$$(a-i)^2 + (b-j)^2 + (c-k_2)^2 \geq (x-i)^2 + (y-j)^2 + (z-k_2)^2.$$

两个不等式相加, 得到  $(z-k_1)^2 + (c-k_2)^2 \geq (c-k_1)^2 + (z-k_2)^2$ , 由已知条件  $k_1 < k_2$  可得  $c \leq z$ .  $\square$

如果已知  $V(i, j, k) = (x, y, z)$ , 由命题 2, 对其他二维图像上与  $(i, j, k)$  具有相同行列位置的像素  $(i, j, l)$ , 若  $l < k$ , 则  $V(i, j, l)$  只能在第  $1 \sim z$  个二维图像上; 若  $l > k$ , 则  $V(i, j, l)$  只能在第  $z \sim n$  个二维图像上. 推而广之, 若  $n$  个具有相同行列位置的像素被  $d \left( 1 \leq d \leq \frac{n}{2} \right)$  等分, 则这些像素的最近黑点  $V$  也被分成  $d$  个部分, 求不同等分像素的最近黑点只需到  $V$  的相应部分去寻找即可.

由此, 我们可以采用等分法缩小对最近黑点的搜索范围. 首先, 对第  $\frac{n}{2}$  个二维图像中的像素的最近黑点的处理实际上是对  $n$  个二维图像进行二等分, 应用命题 2, 为寻找它们的最近黑点, 需参与距离计算与比较的二维图像个数可大大减少; 然后寻找两个等分中心图像, 即第  $\frac{n}{4}$  和第  $\frac{3n}{4}$  个二维图像中像素的最近黑点, 处理结束后,  $n$  个二维图像被四等分; 然后寻找 4 个等分中心图像, 即第  $\frac{n}{8}, \frac{3n}{8}, \frac{5n}{8}$  和  $\frac{7n}{8}$  个二维图像中像素的最近黑点; 依此类推, 直到最后寻找  $\frac{n}{2}$  个等分中心图像中的像素的最近黑点为止. 对  $n$  个二维图像进行  $1, 2, 4, 8, \dots, \frac{n}{2}$  等分, 共进行  $\log n$  次等分后, 每一个二维图像全部被处理过一次, 则所有像素的最近黑点全被找到.

在上述过程中, 等分越细, 寻找等分中心二维图像上像素的最近黑点的搜索范围越小, 即参与距离计算与比较的二维图像个数越少.

**命题 3.** 设  $n$  个二维图像已被  $d \left( d=1, 2, 4, 8, \dots, \frac{n}{2} \right)$  等分, 为寻找  $d$  个等分中心二维图像中相同行列位置的一组像素的最近黑点, 需要参与距离计算的次数小于  $1.5n$ .

证明:  $n$  个具有相同行列位置的像素被  $d$  等分, 根据命题 2, 这些像素的最近黑点  $V$  也被分成  $d$  个部分, 设其中第  $k$  部分的元素个数为  $l_k$ , 则  $\sum_{k=1}^d l_k = n$ , 由于  $V$  在相邻两部分的边界上的元素同时属于两个部分, 为了求距离第  $k$  个等分的中心像素最近的黑点, 需要使用  $V$  的相应部分的  $l_k + 1$  个元素进行距离计算和比较. 对于  $d$  个等分的所有中心像素, 求它们的最近黑点需要  $\sum_{k=1}^d (l_k + 1) - 1 = n + d - 1 < 1.5n$  次距离计算.  $\square$

应用命题 3, 求  $d$  个等分中心二维图像中所有像素的最近黑点, 需要参与距离计算的次数小于  $1.5n^3$ .

## 2.2 算法及复杂度分析

下面, 我们给出二维完全欧氏距离变换的算法.

(1) 对  $n$  个二维二值图像进行二维距离变换, 求得每个像素  $(i, j, k)$  在其所在二维图像中的最近黑点, 记为  $M(i, j, k)$ ;

(2) 求第  $\frac{n}{2}$  个二维图像中像素在三维空间中的最近黑点, 其中每个像素  $(i, j, n)$  必须与  $n-1$  个点  $M(i, j, k)$  ( $k=1, 2, \dots, n-1$ ) 进行距离计算与比较;

(3) for (int  $\lambda=0; \lambda \leq \log n - 1; \lambda++$ )

(4) 对  $n$  个二维图像进行  $d=2^\lambda$  等分

- (5) for (int l=1; l<=d; l++)
- (6) 求第  $\left(\frac{2l-1}{2d}\right)n$  个图像在三维空间中的最近黑点,其中每个像素  $\left(i, j, \frac{2l-1}{2d}n\right)$  与其最近黑点所在范围内的点  $M(i, j, k)$  进行距离计算与比较
- (7) endfor
- (8) endfor

因为最近黑点是通过距离比较得到的,因此,每个像素的最近黑点在三维空间中找到后,它们之间的距离也同时被求得。

根据文献[2],一个  $n \times n$  的二维图像,其欧氏距离变换算法的时间复杂度为  $O(n^2)$ ,则第 1 步的时间复杂度为  $O(n^3)$ 。算法的主要计算过程在第 3~8 步,根据命题 3,第 5~8 步需参与距离计算的次数小于  $1.5n^3$ ,结合第(3)步可知,整个算法的时间复杂度为  $O(n^3 \log n)$ 。

### 3 医学 CT 图像内部软组织的显示

对于三维医学 CT 图像,由于软组织的分辨率比较低(不如医学 MR 图像),因此,采用传统的分割方法,如阈值法不易区分出肌肉、神经血管等不同的组织。用直接体绘制方法显示原有的二维数据,由于受表皮及皮下脂肪的影响,也无法真实地显示出内部的肌肉、神经血管等软组织。我们采用第 2 节所描述的欧氏距离变换算法对原始数据进行预处理,先剥去外层的表皮及皮下脂肪,然后对处理过的数据进行体绘制<sup>[6]</sup>,可得到很好的显示效果。

我们在此仅简单叙述医学 CT 图像内部软组织显示的处理过程。对一个三维医学 CT 图像,首先将目标物体从背景中分离出来,并以背景点为特征点,其余部分为非特征点(为顾及物体中空,我们只提取物体外轮廓以外的背景点为特征点),形成三维二值图像。然后对此三维二值图像进行三维距离变换,得到一个距离图像,距离图像中每一个非特征点的值表示该点到物体外轮廓的最近距离。在此距离图像的基础上,结合表皮及皮下脂肪的 CT 值,我们对原始的医学 CT 图像进行一定深度的由物体外轮廓向内的剥皮处理,将表皮及皮下脂肪剥去得到新的数据,然后对此数据进行三维直接体绘制。我们在 Pentium II 300 的微机上实现了上述算法。实验数据为人体头部 CT 图像,共 58 个切片,间距 1.5mm,分辨率为  $512 \times 512 \times 16$  位,实验结果如图 1 所示。其中,图 1 中(a)为原始数据中的一个切片图像;(b)为算法生成的距离图像;(c)和(d)为用不同的距离深度对原始图像进行剥皮处理并进行三维体绘制生成的结果图像,从中可以清楚地显示骨骼、肌肉及神经血管(在脸颊及下颌位置)等组织。

### 4 小 结

本文在二维欧氏距离变换算法的基础上提出了一种三维欧氏距离变换的新方法,其时间复杂度为  $O(n^3 \log n)$ 。我们将该方法应用于三维医学 CT 图像内部软组织的显示,取得了较好的效果。我们今后的工作将致力于使三维欧氏距离变换算法的时间复杂度降至  $O(n^3)$ ,并考虑将三维欧氏距离变换算法应用于三维数据场的直接体绘制,以降低传统的体绘制方法所需的大量处理时间。

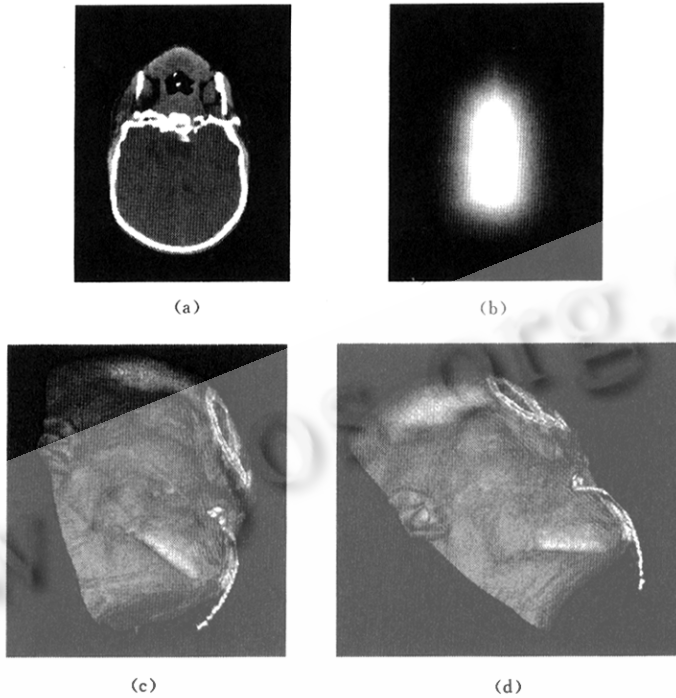


Fig. 1 Display of three-dimensional soft tissue  
图 1 三维软组织显示

### References :

- [1] Rosenfield, A., Pfaltz, J. Sequential operations in digital picture processing. *Journal of Association for Computing Machinery*, 1996,13(4):471~494.
- [2] Chen, Ling. Optimal algorithm of true Euclidean distance transform. *Chinese Journal of Computers*, 1995,8(18):611~616 (in Chinese).
- [3] Okabe, N., Toriwaki, J., Fukumura, T. Paths and distance functions on three-dimensional digitized pictures. *Pattern Recognition Letter*, 1983,1:205~212.
- [4] Borgefors, G. On digital distance transforms in three dimensions. *Computer Vision and Image Understanding*, 1996,64(3):368~376.
- [5] Ragnemalm, I. The Euclidean distance transform in arbitrary dimensions. *Pattern Recognition Letter*, 1993,14:883~888.
- [6] Udupa, J.K., Odhner, D. Shell rendering. *IEEE Computer Graphics & Application*, 1993,13(6):58~67.

### 附中文参考文献:

- [2] 陈陵. 完全欧几里德距离变换的最优算法. *计算机学报*, 1995,8(18):611~616.

## A New Method of Three-Dimensional Euclidean Distance Transform\*

ZHUGE Ying<sup>1</sup>, TIAN Jie<sup>1</sup>, WANG Wei-hong<sup>2</sup>

<sup>1</sup>(AI Laboratory, Institute of Automation, The Chinese Academy of Sciences, Beijing 100080, China);

<sup>2</sup>(Department of Electronic Engineering, Beijing Institute of Technology, Beijing 100081, China)

E-mail: tian@doctor.com

http://www.3dmed.net

**Abstract:** Most existing algorithms of three-dimensional distance transform are extensions of two-dimensional approximate Euclidean distance transform algorithms such as the city block/chessboard. Such algorithms can only get the approximate Euclidean distance. A new method of three-dimensional true Euclidean distance transform is presented in this paper. The proposed method can get the true Euclidean distance with time complexity  $O(n^3 \cdot \log n)$ . Moreover, this method is used to render the soft tissue in three-dimensional medical CT images, and good result has been obtained.

**Key words:** distance transform; complexity; soft tissue rendering

\* Received May 4, 1999; accepted December 27, 1999

Supported by the National Natural Science Foundation of China under Grant No. 69843001; the National High Technology Development Program of China under Grant No. 863-305-ZD04-06-4

**Call for Papers**

**2001 International Conference on Computer Networks and Mobile Computing (IccNMc-01)**

October 16~19, 2001, Beijing, China

Sponsored by China Computer Federation, in cooperation with IEEE Computer Society Beijing Center and IEEE Computer Society Technical Committee on Distributed Processing.

**Scope of Conference** The conference provides a forum for engineers and scientists in academia, industry and government to present their latest research findings in any aspects of computer networks and mobile computing. Topics of interest include, but are not limited to;

- Wireless Networks
- Mobile Computing
- Multimedia Systems
- Intetnet and Web Applications
- Optical Networks
- Protocol Design and Analysis
- Admission/Congestion/Flow control
- Quality of Service
- Routing and Scheduling
- Network Security and Privacy

**Submission of Papers** Form of Manuscript: Not to exceed 25 double-spaced, 8.5X11-inch pages (including figures, tables and references) in 10~12 point font. Number each page. Include an abstract, five to ten keywords, the technical area (s) mst relevant to your paper, and the corresponding author's E-mail address.

- **Electronic Submission:** Web-Based submissions are strongly encouraged. Please see the conference web page for details.
- **Postal Submission:** If electronic submission is not possible, please send s.x hard copies of the manuscript to:

Professor LI Xiaoming  
 Department of Computer Science and Technology, Peking University  
 Beijing, 100871, China  
 E-mail: lxm@pku.edu.cn Fax: +86-10-3275 5813

**Important Dates**

- Submission Deadline: May 1, 2001
- Author Notification: July 1, 2001
- Final Manuscript Due: August 1, 2001

For further information, please visit the conference web sites; <http://conference.pku.edu.cn/iccnmc>; [www.cis.ohio-state.edu/~iccnmc](http://www.cis.ohio-state.edu/~iccnmc), or contact; [shizz@ics.ict.ac.cn](mailto:shizz@ics.ict.ac.cn) (Shi Zhong-zhi); [lu@cis.ohio-state.edu](mailto:lu@cis.ohio-state.edu) (Liu Ming T.).