

# 数字图象处理技术讲座

## 第五讲 计算机断层 (CT) 技术

刘 政 凯

(中国科学技术大学无线电电子学系)

### 一、计算机断层技术原理

大家知道,计算机断层 (CT) 技术的发明是生物医学探测上的一次伟大革命,也是数字图象处理技术在生物医学上的成功应用.CT 可应用于人体脑部及全身各部分的扫描,因而可呈现出人体内部结构的图象.此外,它还可对身体内部的本底吸收和辐射模式进行定量的估值.因此,CT 在功能方面和病理学上具有重大意义.从目前水平上看,与当前的辐射医学和放射医学的诊断方法相比,CT 可检测和定位更小的病灶.因此,CT 在对疾病的早期诊断和处理上将有重大的改善.此外,CT 技术是无损伤探测.有人统计目前美国大约有 2000 台 X 光 CT 设备.预计 CT 技术将会有更大的发展.

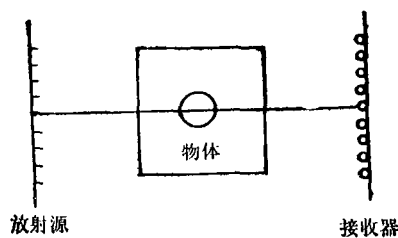


图 1 CT 原理示意图

X 射线的 CT 设备工作原理如图 1 所示. X 光放射源和接收器分别放于病人的两侧.放射源发出一束 X 光,穿过病人的内部器官,由于不同器官及不同特性使得光束受到不同程度的吸收,因此在接收器上接收到的 X 光的强度也

不同,这时便可得到某一方位的投影.把放射源和接收器同步旋转另一方位角,便可得到另一组投影,这样继续下去便可获得无穷多个投影.根据这些投影便可重建出病人内部结构的断层象.

放射源大致有三种<sup>[1]</sup>:

1. 平行源: 在平行源同一平面上发射出平行的射线束.
2. 扇形源: 在扇形源的同一直线上发射出发散的射线束.
3. 锥形源: 锥形源在空间上发射一锥形射线束.

CT 中的重建问题,就是根据 CT 扫描获得的有限的投影,对物体断面结构分析作出最好的估值.由于投影数据量非常之大,因此研究有效的重建方法是非常重要的.目前已研究出不少 CT 图象重建的算法,大体上可分为直接法和间接法.直接法是利用原始投影数据在空域中进行计算;间接法是在原始空域的变换域中进行运算.具体可分五类:

1. 相加法<sup>[2]</sup>: 在各个投影上,把通过物体某一点的射线强度的直接相加,即可得到该点密度的估值.对断层面上每一点都进行这样的相加运算后,即可得到整幅的 CT 断层象.例如,图 2(a)是由二个投影,采用相加法重建出的物体断层象;图 2(b)是由三个投影重建象;图 2(c)是由四个投影的重建象.由图可见,投影愈多,重建愈精确.

2. 傅里叶变换法: 把各个方位角的投影数据进行傅里叶变换,便得到整幅断层象的傅里

叶变换样本值。样本间点的数值可用内插法得到。然后进行反变换,即可得到断层图。

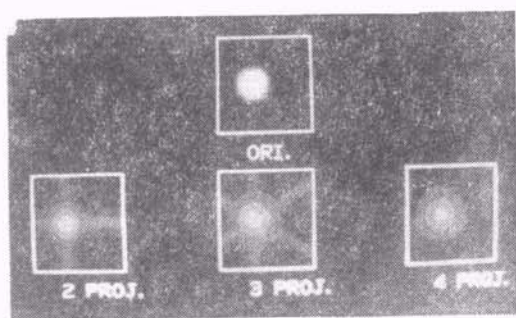


图2 相加法重建象

3. 解积分方程组方法: 根据 CT 成象模型可得出, 断层象与投影数据之间的基本关系可用一组积分方程表示, 解这个方程组便可得到断层象。

4. 级数展开法: 断层象的估值可利用预先确定的基本图象的线性组合得到, 而线性组合的系数可从把未知图象的投影表示为基本图象的线性组合的方程组中得到。

5. Bayes 估值。给出投影数据后, 根据 Bayes 估值方法可求出最大似然估值。

当前, 改善 CT 技术的方向除了上面谈到的改进重建算法以外, 还致力于提高扫描速度。因为在扫描时间内, 病人的移动会造成断层象的严重模糊。一个有效的办法是采用扇形放射源, 并用一阵列接收器进行接收。这样可把成象时间缩短到几秒之内, 并可利用计算机实时成象。在几秒钟的时间内, 病人有可能保持不动, 但对心脏的跳动, 几秒钟还是显得太长了。因此, 还需努力提高扫描速度, 缩短成象时间。

## 二、频域恢复法<sup>[3]</sup>

设  $g(x, y)$  代表一个二维图象, 射线发射源为平行射线射源。射线  $AB$  穿透图象  $g(x, y)$ , 如图 3 所示。

在  $x, y$  坐标平面上, 直线  $AB$  可表示为

$$x \cos \theta + y \sin \theta = t_1, \quad (1)$$

其中  $t_1$  代表坐标原点到直线  $AB$  的法线距离,

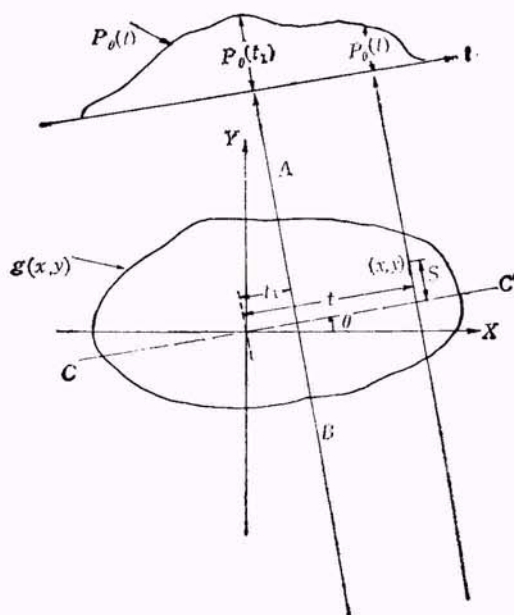


图3 平行投影

$\theta$  为投影法线与  $x$  轴的夹角, 称为倾角。射线  $AB$  穿过图象时形成一个投影  $P_\theta(t_1)$ , 称为射线投影。各条射线投影的总合称为  $\theta$  角投影。射线  $AB$  的投影  $P_\theta(t_1)$  就是图象  $g(x, y)$  沿直线  $AB$  的积分, 并可用下式表示:

$$\begin{aligned} P_\theta(t_1) &= \int_{AB} g(x, y) ds \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(x, y) \delta(x \cos \theta \\ &\quad + y \sin \theta - t_1) dx dy. \end{aligned} \quad (2)$$

各射线投影的总合用  $P_\theta(t)$  表示, 它是射线倾角  $\theta$  及射线法线距离  $t$  的函数。改变倾角, 可以得到一系列投影。图象重建的目标就是, 根据这一系列不同倾角的投影, 重建出原图象  $g(x, y)$ 。

下面介绍频率重建法。

首先图象与其傅里叶变换之间用下式表示:

$$G(U, V) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(x, y) \exp[-j2\pi(Ux + Vy)] dx dy \quad (3)$$

以及

$$g(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} G(U, V) \exp[j2\pi(Ux + Vy)] dU dV. \quad (4)$$

同理,令投影  $P_\theta(t)$  的傅里叶变换为  $S_\theta(\omega)$ , 即

$$S_\theta(\omega) = \int_{-\infty}^{\infty} P_\theta(t) \exp[-j2\pi\omega t] dt. \quad (5)$$

我们首先考虑  $G(U, V)$  在  $V=0$  的直线上的数值为

$$\begin{aligned} G(U, 0) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(x, y) \exp[-j2\pi Ux] dx dy \\ &= \int_{-\infty}^{\infty} \left[ \int_{-\infty}^{\infty} g(x, y) dy \right] \\ &\quad \cdot \exp(-j2\pi Ux) dx \\ &= \int P_0(t) \exp(-j2\pi Ut) dt \\ &= S_0(U), \end{aligned} \quad (6)$$

其中  $\int_{-\infty}^{\infty} g(x, y) dy$  即是  $\theta=0$  的投影  $P_0(t)$ .

由(6)式可以得出这样的结论:  $G(U, V)$  在  $V=0$  直线上的数值即是  $\theta=0$  的投影  $P_0(t)$  的傅里叶变换. 同理, 如果用  $G(\omega, \theta)$  代表  $G(U, V)$  在以  $V$  轴成  $\theta$  角上的数值, 则可得

$$G(\omega, \theta) = S_\theta(\omega). \quad (7)$$

从以上推导可以看出频域法图象重建的思想为: 根据各投影的傅里叶变换, 可求出原图象的富氏变换, 再进行傅里叶反变换, 就可重建出原图象. 数学表示式如下:

$$\begin{aligned} g(x, y) &= \int_0^{2\pi} \int_0^{\infty} G(\omega, \theta) \exp[j2\pi\omega(x \cos \theta \\ &\quad + y \sin \theta)] \omega d\omega d\theta \\ &= \int_0^{\pi} \int_0^{\infty} G(\omega, \theta) \exp(j2\pi\omega(x \cos \theta \\ &\quad + y \sin \theta)] \omega d\omega d\theta \\ &\quad + \int_0^{\pi} \int_0^{\infty} G(\omega, \theta + 180^\circ) \\ &\quad \times \exp\{j2\pi\omega[x \cos(\theta + 180^\circ) \\ &\quad + y \sin(\theta + 180^\circ)]\} \omega d\omega d\theta. \end{aligned}$$

由于  $G(\omega, \theta + 180^\circ) = G(-\omega, \theta)$ , 因此上式可改写成

$$\begin{aligned} g(x, y) &= \int_0^{\pi} \left[ \int_{-\infty}^{\infty} G(\omega, \theta) |\omega| \right. \\ &\quad \times \exp(j2\pi\omega t) d\omega \left. \right] d\theta \\ &= \int_0^{\pi} \left[ \int_{-\infty}^{\infty} S_\theta(\omega) |\omega| \right. \\ &\quad \times \exp(j2\pi\omega t) d\omega \left. \right] d\theta. \end{aligned} \quad (8)$$

因此,

$$g(x, y) = \int_0^{\pi} Q_\theta(x \cos \theta + y \sin \theta) d\theta, \quad (9)$$

其中

$$Q_\theta(t) = \int_{-\infty}^{\infty} S_\theta(\omega) |\omega| \exp(j2\pi\omega t) d\omega, \quad (10)$$

或者写成

$$Q_\theta(t) = \int_{-\infty}^{\infty} P_\theta(d) h_1(t - \alpha) d\alpha, \quad (11)$$

其中

$$h_1(t) = \int_{-W}^W |\omega| \exp(j2\pi\omega t) d\omega, \quad (12)$$

称为限带系统的等效滤波器的点扩展函数, 这里  $W$  为系统带限.

上述原理实际实现, 可采用以下步骤:

第一步: 滤波. 上述分析都是对连续模拟信号而言. 为便于计算机处理, 首先要数字化, 为此必须采样. 假设对每个投影进行采样, 采样间隔为  $\tau$ . 为了避免混叠失真, 采样间隔应满足  $W = 1/2\tau$ . 把这个数值代入(12)式中, 可得

$$\begin{aligned} h_1(t) &= \frac{1}{2\tau^2} \left( \frac{\sin(2\pi t/2\tau)}{2\pi t/2\tau} \right) \\ &\quad - \frac{1}{4\tau^2} \left( \frac{\sin(\pi t/2\tau)}{\pi t/2\tau} \right)^2. \end{aligned} \quad (13)$$

由于采样间隔为  $\tau$ , 则  $\tau$  可表示成  $n\tau$ , 即

$$h_1(n\tau) = \begin{cases} 1/4\tau^2 & n=0, \\ 0 & n \text{ 为偶数} \\ -(1/n^2\pi^2\tau^2) & n \text{ 为奇数,} \end{cases} \quad (14)$$

其中  $n$  为正、负整数.

在采样点  $n\tau$  处, 根据(11)式, 滤波后的投影值可表示为

$$Q_\theta(n\tau) = \tau \sum_{m=-\infty}^{\infty} P_\theta(m\tau) h_1[(n-m)\tau]. \quad (15)$$

对于带限函数,  $m$  为有限值. 即当  $m=0, \dots, N-1$  时,  $P_\theta(m\tau)$  为有限值; 而当  $m > N-1$  时,  $P_\theta(m\tau)$  为零. 因此, 上式可改写成

$$Q_\theta(n\tau) = \tau \sum_{m=0}^{N-1} P_\theta(m\tau) h_1[(n-m)\tau] \quad (16)$$

或者

$$Q_\theta(n\tau) = \tau \sum_{m=-(N-1)}^{N-1} P_\theta[(n-m)\tau] h_1(m\tau), \quad (17)$$

其中  $n = 0, 1, 2, \dots, N-1$ . 上面二个等式意味着, 为了从  $P_\theta(t)$  的  $N$  个采样值求得  $Q_\theta(n\tau)$ ,  $h_1(n\tau)$  的长度应当从  $n = -(N-1)$  到  $n = (N-1)$ .

虽然 (16) 式和 (17) 式中的离散卷积可以在通用计算机上实现, 但是通常在频域里利用 FFT 算法会更快一些. 为了能在频域中运算, 必须把离散投影值  $P_\theta(m\tau)$  和系统点展函数  $h_1(m\tau)$  的序列长度都适当填零加长, 以保证 (16) 式和 (17) 式为周期卷积.

因此, 频域图象重建方法的第一步就是利用 FFT 算法, 按 (16) 式和 (17) 式计算出  $Q_\theta(n\tau)$ . 即首先取  $P_\theta(m\tau)$  和  $h_1(m\tau)$  填零加长后的序列的快速傅氏变换. 然后把二者的傅氏变换值相乘, 再取傅氏反变换, 即得到  $Q_\theta(n\tau)$ . 由于这一步与通常的滤波算法相似, 所以这一步也称做滤波, 并称  $Q_\theta(n\tau)$  为滤波后投影值.

第二步: 相加. 在第一步中已求出滤波后投影值  $Q_\theta(n\tau)$ , 这一步就是利用 (9) 式求出原图象  $g(x, y)$ . 为此, 也需要把连续积分用数值近似来代替. 假设, 投影数目  $M$  很大并且在  $180^\circ$  内平均分布, 则 (9) 式可近似为

$$\hat{g}(x, y) = \frac{\pi}{M} \sum_{i=1}^M Q_{\theta_i}(x \cos \theta_i + y \sin \theta_i). \quad (18)$$

这个等式可以作如下解释: 当倾角  $\theta$  一定时, 对于给定的各点  $(x, y)$ , 都有一确定的  $t$  值, 即

$$t = x \cos \theta + y \sin \theta.$$

$Q_{\theta_i}$  对点  $(x, y)$  的影响取决于  $t$  值的大小. 因此, 对于图象  $g(x, y)$  上通过点  $(x, y)$ , 并且法线距离为  $t$  的直线上各点,  $Q_{\theta_i}$  对这些点贡献相同. 把通过点  $(x, y)$  的各个滤波后投影对该点的贡献相加, 即可重建出点  $(x, y)$  的数值. 逐点计算即可重建出原图象  $g(x, y)$ .

### 三、代数恢复法

如果把一幅图象  $g(x, y)$  用网格表示, 在每一网格中灰度值为常数. 例如, 用  $x_m$  表示第  $m$  个网格中的灰度值,  $N$  为总的网格数目, 如图

4 所示. 在代数重建方法中, 我们定义射线是具有一定宽度  $\tau$ , 在大多数情况下, 射线宽度  $\tau$  等于网格宽度. 这时称射线积分为射线和.

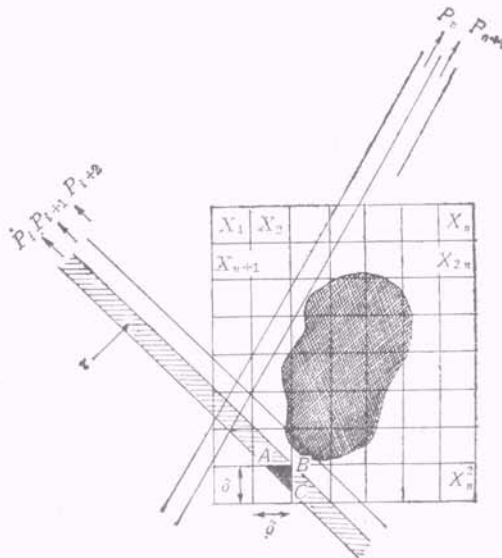


图4 代数恢复法

设第  $i$  条射线的射线和为  $P_i$ , 则  $x_i$  与  $P_i$  的关系可表示为

$$\sum_{j=1}^N w_{ij} x_j = P_i \quad i = 1, 2, \dots, M, \quad (19)$$

其中  $M$  为所有投影的射线总数,  $w_{ij}$  代表第  $i$  个网格对第  $j$  条射线积分的加权因子,  $w_{ij}$  等于第  $j$  条射线在第  $i$  条网格中截取的面积. 显然只有该射线经过的那些网格的  $w_{ij}$  才不为零, 而该射线不通过的那些网格的  $w_{ij}$  为零.

由于  $M$  和  $N$  的数值很大. 因此, 在 CT 中的代数恢复方法中很难用矩阵进行运算. 例如, 对于一幅  $256 \times 256$  的图象,  $N$  可能近似等于 65000,  $M$  也接近这个数值. 用矩阵表示时, 矩阵尺寸为  $65000 \times 65000$ . 这样大的矩阵是很难存贮和运算的.

显然 (19) 式可以写成如下的方程组:

$$\begin{cases} w_{11}x_1 + w_{12}x_2 + w_{13}x_3 + \dots + w_{1N}x_N = P_1, \\ w_{21}x_1 + w_{22}x_2 + w_{23}x_3 + \dots + w_{2N}x_N = P_2, \\ \dots \\ w_{M1}x_1 + w_{M2}x_2 + w_{M3}x_3 + \dots + w_{MN}x_N = P_M. \end{cases} \quad (20)$$

用  $N$  个网格表示一幅图象, 意味着图象具有  $N$  个自由度。  $(x_1, x_2, \dots, x_N)$  可看成  $N$  维空间中的一个点。 在这个空间中, 上式的每一个等式代表  $N$  维空间中的一个超平面。 如果上式存在单一解, 那么这个解就是超平面的交点。

我们已知上述方程组的投影法解法, 即

$$\mathbf{x}^{(j)} = \mathbf{x}^{(j-1)} - \frac{(\mathbf{x}^{(j-1)} \cdot \mathbf{w}_j - p_j)}{\mathbf{w}_j \cdot \mathbf{w}_j} \mathbf{w}_j, \quad (21)$$

其中  $\mathbf{w}_j = (w_{j1}, w_{j2}, \dots, w_{jN})^T$ ,  $\mathbf{x} = (x_1, x_2, \dots, x_N)^T$ 。

为了克服加权因子  $w_{ij}$  在计算、存贮和检索上的困难, 在实际应用(21)式时, 需要作某种近似。 为此, 我们改写上式

$$\mathbf{x}_m^{(j)} = \mathbf{x}_m^{(j-1)} + \left[ (p_j - q_j) / \sum_{k=1}^N w_{jk}^2 \right] w_{jm}, \quad (22)$$

其中

$$\begin{aligned} q_j &= \mathbf{x}^{(j-1)} \cdot \mathbf{w}_j \\ &= \sum_{k=1}^N x_k^{(j-1)} w_{jk}. \end{aligned} \quad (23)$$

这些等式意味着, 当我们把第  $(j-1)$  次解投影到第  $j$  个超平面上, 并且当前灰度值为  $x_m^{(j-1)}$  时的第  $m$  个元素的灰度修正值  $\Delta x_m^{(j)}$  为

$$\begin{aligned} \Delta x_m^{(j)} &= x_m^{(j)} - x_m^{(j-1)} \\ &= \frac{p_j - q_j}{\sum_{k=1}^N w_{jk}^2} w_{jm}. \end{aligned} \quad (24)$$

上式说明,  $p_j$  为第  $j$  条射线的测量射线和,  $q_j$  为该射线基于第  $j-1$  次解的计算和。 第  $m$  个网格的校正值  $\Delta x_m$  取决于射线测量和与计算和之差。

简单近似方法之一是把  $w_{jk}$  用 0 和 1 代替, 即如果该网格中心位于第  $j$  条射线之内, 则  $w_{jk} = 1$ , 否则  $w_{jk} = 0$ 。 这种近似法称为近似代数重建法 (ART)。 显然, 这时  $\sum_{k=1}^N w_{jk}^2 = N_j$ 。

其中  $N_j$  为其中心位于第  $j$  条射线上的图象网格数目。 这时 (24) 式可以改写为

$$\Delta x_m^{(j)} = (p_j - q_j) / N_j. \quad (25)$$

根据该公式可以很容易修正位于第  $j$  条射线上的

的网格灰度值。

(25) 式计算简单, 易于实现, 但往往在重建图象上造成某些人为的干扰条纹。

另一种近似方法, 是令

$$\Delta x_m^{(j)} = p_j / L_j - q_j / N_j, \quad (26)$$

其中  $L_j$  为第  $j$  条射线穿透重建区域的归一化长度 (即按照网格宽度  $\delta$  归一化)。

根据上面介绍的近似公式, 便可以重建出 CT 图象。 以上的近似投影法称为 ART (代数重建技术) 算法。

ART 算法的优点是收敛性好。 缺点是重建的图象存在大量的呈胡椒面样式的噪声。

为了消除胡椒面式噪声而得到平滑图象, 可采用同时迭代重建技术 (SIRT) 算法。 这种算法的基本思想首先按照 (25) 或 (26) 式对 (20) 式中的所有等式计算出  $\Delta x_m$ , 然后取其平均值去更新该象素的灰度值。 从而实现了第一次迭代过程。 然后, 从 (20) 式中第一个等式开始第二次迭代过程。 如此重复下去, 直到得到满意的重建图象。 该算法可以得到较平滑的图象, 但收敛性稍差。

在 ART 和 SIRT 算法中, 胡椒面式噪声的造成是由于 (20) 式中各项加权系数  $w_{jk}$  只是简单用 0 和 1 近似而造成的。 为了克服上述两种算法的缺点, 提出了一种称为 SART (同时代数重建技术) 算法。

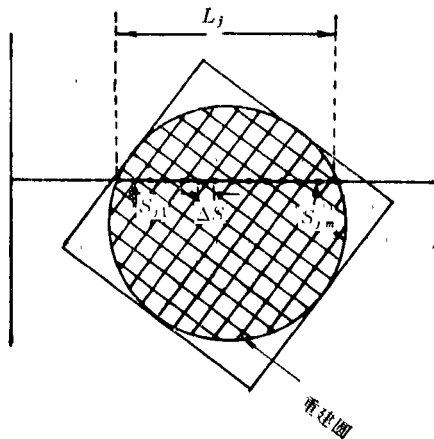


图 5 SART 算法

SART 算法的思想是, 假定需要重建的目

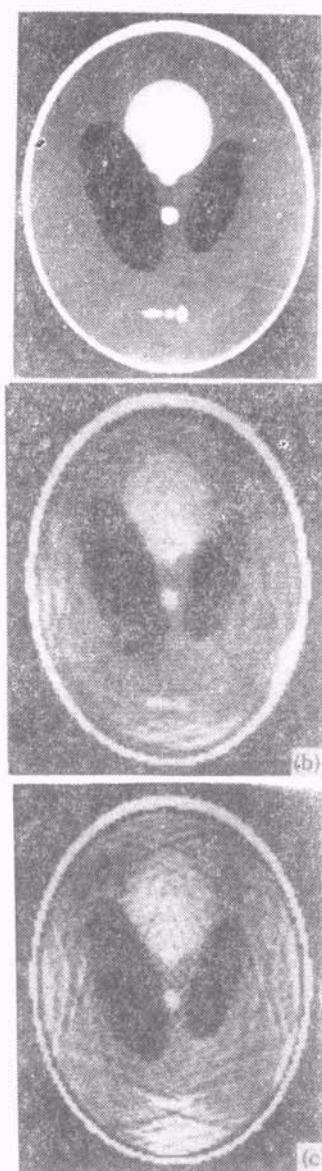


图6 用 SART 算法重建的图象

标位于一个重建圆内,如图5所示。重建圆仍然画分成  $\delta \times \delta$  的网格。

根据图5,可以把(19)式改写成<sup>[4]</sup>

$$p_j = \sum_{m=1}^{p_j} g(s_{jm}) \Delta s, \quad (27)$$

其中  $s_{jm}$  是图5中在  $xy$  平面上第  $j$  条射线上的等间隔点,  $p_j$  是该射线上总的点数。  $g(s_{jm})$  代表  $s_{jm}$  点上的灰度值,它可以根据其周围的四个象元的灰度值  $x_n$  通过内插而得到。因此,通常可写成

$$g(s_{jm}) = \sum_{i=1}^N d_{ijm} x_i. \quad (28)$$

在双线性内插时,对于每一个  $s_{jm}$ ,只是最靠近的四个点  $d_{ijm}$  不为零,而且其总和等于1。根据以上两个等式,我们得到

$$p_j = \sum_{i=1}^N w_{ij} x_i \quad (29)$$

和

$$w_{ij} = \sum_{m=1}^{p_j} d_{ijm} \Delta s. \quad (30)$$

显然,每一射线上从起点到终点的象素的加权因子之和必然等于射线实际长度  $L_j$ ,即

$$\sum_{i=1}^N w_{ij} = L_j.$$

实验证明,如果取  $\Delta s = \frac{1}{2} \delta$  该 SART 算法在

精度和计算时间之间具有较好的折衷。

加权因子选好后,采用投影法,即可重建出质量较好的图象。图6(a)为原图,图6(b)为采用第一次迭代的  $128 \times 128$  重建象。在计算机模拟 CT 数据时,在  $180^\circ$  内采用100个平行投影,每个投影中包含127条射线。图6(c)为经过二次迭代的重建象。

### 参 考 文 献

- [1] K. Preston, Jr, and M. Onoc, Digital Processing of Biomedical Images, Plenum Press, (1976).
- [2] 傅京孙著,模式识别及其应用,科学出版社,(1983).
- [3] M. P. Ekstrom, Digital Image Processing Techniques, Academic Press, INC., (1984).
- [4] A. H. Andersen and A. C. Kak., Ultrasonic Imaging, No. 6, (1984), 81.