样条函数在图像处理和传输中的应用

王　绍　霖

提　要　样条函数由于其计算简单，便于数字计算机进行数值计算而在各个工程领域内广泛地得到应用。在连续图像的离散化数字处理中，由于图像数据量非常巨大，使用样条函数进行内插和平滑可以得到计算简化的好处。本文初步讨论这方面应用的可能性。本文所述及的范围还比较小，只是试论用样条函数来估计图像离散化和再现过程中的误差；用样条函数进行数据压缩的一种简便方法；和用样条函数对图像随机噪声进行平滑处理等几个方面。

一、样条函数简述

在用数字计算机对连续的（无限維的）问题求解时，必须用离散化的方法（化为有限維的），即首先把无限維问题化为有限維问题。在把連续的问题变成离散的问题以求其计算上的近似解时，采用变分和逐段多项式相结合的方法是很成功的，在工程技术上人们称之为有限元法。有限元法的应用范围是很普遍的，它的适应性强，形式单纯，规范化，在许多物理和力学问题中得到成功的应用。而逐段多项式中的一种——样条函数在解的光滑性和计算的简单性方面有较大的优点。本节先介绍一些简单的有关知识，为以后几节的分析作准备。

逐段多项式内插是解有限維问题的一个有效方法，逐段多项式 $u_i=a_{i1}+a_{i2}x+a_{i3}x^2+\cdots+a_{in+1}x^n$ 为 n 次多项式，表示 x_i 和 x_{i+1} 结点之间的内插曲线，进行样条内插时，已知区间 $\Delta: 0=x_0<x_1<\cdots<x_{N+1}=1$ 内的离散点 $x_0,x_1,\cdots,x_i,\cdots,x_{N+1}$ 上的函数值 $\boldsymbol{f}=\{f_0,f_1,\cdots,f_i,\cdots,f_{N+1}\}$ 和 f_0',f_{N+1}'；f_0' 和 f_{N+1}' 为区间两端的函数导数值，以样条函数来逼近函数 f. 以三次样条函数为例，三次样条是最常用的，它的二阶以下导数在结点处（分段节点处）保持連续性。这样一种連续性要求，使系数 $a_{i1},a_{i2},\cdots$ 等有一定的约束，即各段的三次曲线不是任意曲线，而是具有一定形状的曲线，例如Hermite多项式以 $h(x_i)$ 表示，为使它们在结点处保持連续，样条内插式可表示为

$$\hat{f}=\sum_{i=0}^{N+1} f_i h_i(x)+f_0' h_0^1(x)+\sum_{i=1}^{N} c_i^1 h_i^1(x)+f_{N+1}' h_{N+1}^1(x) \tag{1}$$

(1)式中的 c_i^1 可从(2)式中解出来

$$[B]\boldsymbol{C}^1=\boldsymbol{K} \tag{2}$$

Hermite 多项式 $h_i(x)$ 可以表示为

本文1982年9月22日收到

$$h_0(x)=\begin{cases}2x_1^{-3}x^3-3x_1^{-2}x^2+1 & 0\leqslant x\leqslant x_1\\ 0 & x_1\leqslant x\leqslant 1\end{cases}$$

$$h_i(x)=\begin{cases}-2(x_i-x_{i-1})^{-3}(x-x_{i-1})^3+3(x_i-x_{i-1})^{-2}(x-x_{i-1})^2 & x_{i-1}\leqslant x\leqslant x_i\\ 2(x_{i+1}-x_i)^{-3}(x-x_i)^3-3(x_{i+1}-x_i)^{-2}(x-x_i)^2+1 & x_i\leqslant x\leqslant x_{i+1}\\ 0 & x\in[0,1]-[x_i,x_{i+1}]\end{cases}$$

$$h_{N+1}(x)=\begin{cases}-2(1-x_N)^{-3}(x-x_N)^3+3(1-x_N)^{-2}(x-x_N)^2 & x_N\leqslant x\leqslant 1\\ 0 & 0\leqslant x\leqslant x_N\end{cases}$$

其图形如图1所示.

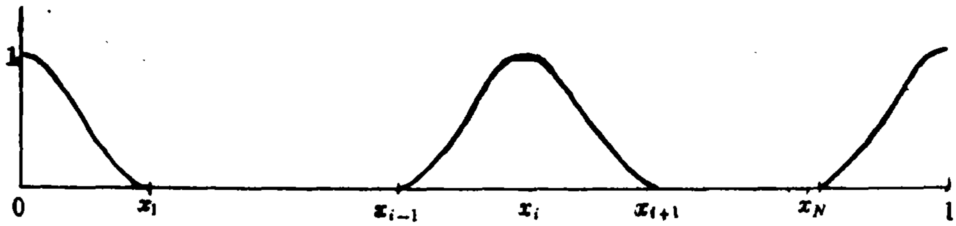

图 1

$$h_0^1(x)=\begin{cases}x_1^{-2}x(x_1-x)^2 & 0\leqslant x\leqslant x_1\\ 0 & x_1\leqslant x\leqslant 1\end{cases}$$

$$h_i^1(x)=\begin{cases}(x_i-x_{i-1})^{-2}(x-x_{i-1})^2(x-x_i) & x_{i-1}\leqslant x\leqslant x_i\\ (x_{i+1}-x_i)^{-2}(x-x_i)(x_{i+1}-x)^2 & x_i\leqslant x\leqslant x_{i+1}\\ 0 & x\in[0,1]-[x_{i-1},x_{i+1}]\end{cases}$$

$$h_{N+1}^1(x)=\begin{cases}(1-x_N)^{-2}(x-x_N)^2(x-1) & x_N\leqslant x\leqslant 1\\ 0 & 0\leqslant x\leqslant x_N\end{cases}$$

其图形如图2所示。

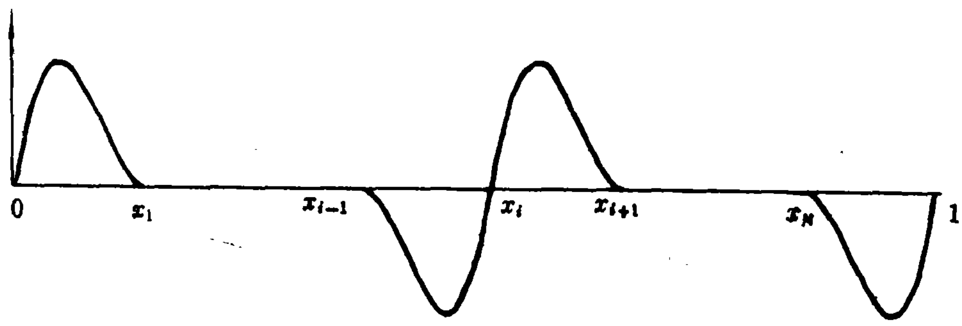

图 2

(2) 式中

$$b_{ij}=\begin{cases}2(\Delta x_i+\Delta x_{i-1}) & 1\leqslant j=i\leqslant N\\ \Delta x_i & 1\leqslant j=i-1\leqslant N-1\\ \Delta x_{i-1} & 2\leqslant j=i+1\leqslant N\\ 0 & \text{其它}\end{cases}$$

其中 $\Delta x_i = x_{i+1} - x_i$; $\Delta c_j = c_{j+1} - c_j$.

$$k_i = \begin{cases} 3[\Delta x_0(\Delta x_1)^{-1}\Delta c_1 + \Delta x_1(\Delta x_0)^{-1}\Delta c_0] - \Delta x_i f_0' & i=1 \\ 3[\Delta x_{i-1}(\Delta x_i)^{-1}\Delta c_i + \Delta x_i(\Delta x_{i-1})^{-1}\Delta c_{i-1}] & 1<i<N \\ 3[\Delta x_{N-1}(\Delta x_N)^{-1}\Delta c_N + \Delta x_N(\Delta x_{N-1})^{-1}\Delta c_{N-1}] - \Delta x_{N-1} f_{N+1}' & i=N \end{cases}$$

$\hat{f}(x)$ 本身是一个分段的三次多项式 $\hat{f}(x_i)=f(x_i)$，令 $c_i=\hat{f}(x_i)$；由上面可以看到 $[B]$ 是三对角矩阵。方程 (2) 有唯一解 c_i^1. 现在按 (1) 式 $\hat{f}(x)$ 表示为以基函数为基础展开的分段多项式，参数适当地减少了，这样有利于计算。

$\hat{f}(x)$ 对 $f(x)$ 的内插误差 $f-\hat{f}$ 的 L^2 范数是对 f 四阶逼近的，如 f 是四次连续可导，在结点上 $(x_1, x_2 \cdots\cdots x_i \cdots)$ 误差 $f-\hat{f}=0$，而在每一分段内其误差界为

$$\|f-\hat{f}\|_2 \leqslant 4\pi^{-4}h^4\|D^4 f\|_2 \tag{3}$$

式中 $\|\cdot\|_2$ 为 L^2 范数，可表示为

$$\|D^4 f\|_2 = \left(\sum_{i=0}^{N}\int_{x_i}^{x_{i+1}} |D^4 f(x)|^2 dx\right)^{\frac{1}{2}}$$

对二維的情况有

$$\|f-\hat{f}\|_2 \leqslant 4\pi^{-4}\rho^4(\|D_x^4 f\|_2 + \|D_x^2 D_y^2 f\|_2 + \|D_y^4 f\|_2) \tag{4}$$

$D=\frac{d}{dx}$ 为导数运算，(3) 式和 (4) 式中，h 和 ρ 为间隔距离，即离散点的间隔，这里假定它们是等间隔的，ρ 为正方形二維网络的距离。

由 (3) 式可见样条函数内插是一种光滑的曲线内插，这是和富氏级数展开不同的。基底函数的选择可使计算方便，其中的一种称为 B 样条或称 δ 样条，它是从 δ 函数的离散近似函数中导出的，它和 $h(x)$ 那样，是一种固定形状的函数。

δ 函数可以看成是单位阶跃函数 $\sigma(x)$ 的广义导数。对离散的情况以差商 $\frac{\Delta h}{h}$ 代替导数 $\frac{d}{dx}$，$\frac{\Delta_h}{h}\approx\frac{d}{dx}$，$\Delta_h f(x)=f\left(x+\frac{h}{2}\right)-f\left(x-\frac{h}{2}\right)$. 由于

$$\frac{d}{dx}\int f(x)dx = f(x)$$

因此

$$f_h(x) = \frac{\Delta_h}{h}\int f(x)dx \approx \frac{d}{dx}\int f(x)dx = f(x)$$

把上述方法用到 δ 函数上，得离散 δ 函数 δ_h

$$\delta_h(x) = \frac{\Delta_h}{h}\int_{-\infty}^{\infty}\delta(x)dx = \frac{\Delta_h}{h}\sigma(x)$$

$$= \frac{\sigma\left(x+\frac{h}{2}\right)-\sigma\left(x-\frac{h}{2}\right)}{h} \approx \delta(x)$$

从图形上看如图3所示，把 $\delta_h(x)$ 再积分一次，再差商一次，就能得到更光滑的函数，它还是 δ 函数的近似。例如

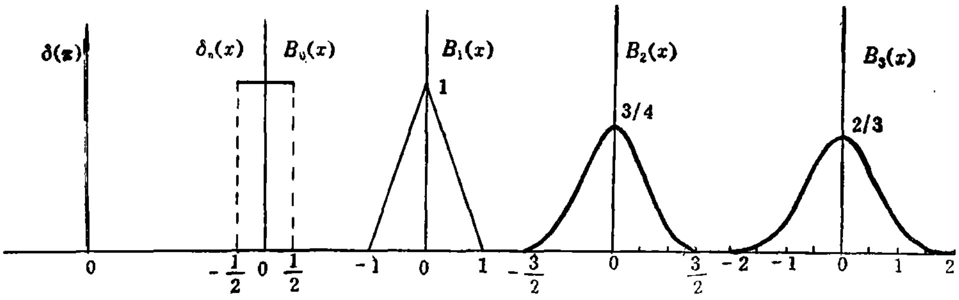

图 3

$$\delta'_h(x)=\frac{\Delta_h}{h}\int_{-\infty}^{x}\delta_h(x)dx=\frac{1}{h^2}[(x+h)_+-2x_++(x-h)_+]$$
$$\approx\delta_h(x)\approx\delta(x)$$

式中

$$(x+h)_+=\begin{cases}(x+h) & x>-h\\ 0 & x<-h\end{cases}$$

对 $\delta(x)$ 作 $(k+1)$ 次积分再作 $(k+1)$ 次差商得到下列函数，这里设 $h=1$.

$$B_k(x)=\Delta^{k+1}\underbrace{\int\cdots\int}_{k+1}\delta(x)dx\cdots dx=\Delta^{k+1}\left(\frac{x_+^k}{k!}\right)$$
$$=\sum_{j=0}^{k+1}\frac{(-1)^j\binom{k+1}{j}\left(x+\frac{k+1}{2}-j\right)_+^k}{k!}\tag{5}$$

这 $B_k(x)$ 相当于 $\delta_h^{(k)}$，它保持了 δ 函数的特性，但已经使它光滑化，去掉了不連续点。最常用的是 $k=3$ 的三次样条函数

$$B_3(x)=\frac{1}{6}(x+2)_+^3-\frac{2}{3}(x+1)_+^3+x_+^3-\frac{2}{3}(x-1)_+^3+\frac{1}{6}(x-2)_+^3\tag{6}$$

$$=\begin{cases}\frac{1}{6}(2-x)^3-\frac{2}{3}(1-x)^3-x^3+\frac{2}{3}(1+x)^3 & -2\leqslant x\leqslant -1\\ \frac{1}{6}(2-x)^3-\frac{2}{3}(1-x)^3-x^3 & -1\leqslant x\leqslant 0\\ \frac{1}{6}(2-x)^3-\frac{2}{3}(1-x)^3 & 0\leqslant x\leqslant 1\\ \frac{1}{6}(2-x)^3 & 1\leqslant x\leqslant 2\\ 0 & 其它\end{cases}$$

形状如图3所示。

不等间隔分布的δ基底函数要复杂一些，这里不涉及，方法类推。

B基底函数有一些特点，简述如下：

1） $B_k(x)$ 是 k 次等间隔样条（k 次分段多项式），其结点为：

$$\xi_j^{(k)}=-\frac{k+1}{2}+j, \qquad j=0,1,\cdots,k+1$$

2） $B_k(x)$ 为对称峯状函数，在 $x=0$ 处是峯值，$x>0(<0)$时单调下降

$$|x|>\frac{k+1}{2}\text{时 } B_k(x)=0$$

3） $\int_{-\infty}^{\infty} B_k(x)dx=1$ 或 $\sum_{n=-\infty}^{\infty} B_k(x+n)=1$

4） $B_k(x)=\int_{-\infty}^{\infty} B_{k-1}(x-t)B_0(t)dt=B_{k-1}(x)\circledast B_0(x)$

或 $B_k(x)=\int_{-\infty}^{\infty} B_{k-n}(x-t)B_{n-1}(t)dt=B_{k-n}\circledast B_{n-1}$

这个关系式可以从δ函数的特性来证明。

5） $B_k'(x)=B_{k-1}\left(x+\frac{1}{2}\right)-B_{k-1}\left(x-\frac{1}{2}\right)$

二、应用的几个方面

在图像的数字处理和传输中，往往要对数字图像数据用一组正交函数进行展开，以求得数值计算上或使用上的简化，而又能保持误差最小，用样条函数展开一个数字图像是一种二維的展开，它具有与其它正交展开不同的特点。在某些場合下可以得到一些有用的效果，由于在这方面进行的工作还很少，本文只就以下几个方面进行一些探讨。

1） 多项式內插

在数字图像处理中，首先要把二維連续图像函数$f(x,y)$进行采样变为离散化图像，而

$$f(x,y)=\sum_m\sum_n f(m\varDelta x,n\varDelta y)g(x-m\varDelta x,y-n\varDelta y) \tag{7}$$

上式中 $f(m\varDelta x,n\varDelta y)$ 是在网络点 $m\varDelta x,n\varDelta y$ 上函数 $f(x,y)$ 的采样值，而 $g(x,y)$是內插函数，按照采样定理，如 $f(x,y)$ 是频谱受限的，即如 $f(x,y)$ 的二維富氏变换 $F(\omega_x,\omega_y)$，当 $|\omega_x|>\omega_{xc}$、$|\omega_y|>\omega_{yc}$ 时 $F(\omega_x,\omega_y)=0$，则当內插函数

$$g(x,y)=\frac{K\omega_{xL}\omega_{yL}}{\pi^2}\cdot\frac{\sin(\omega_{xL}x)}{\omega_{xL}x}\cdot\frac{\sin(\omega_{yL}y)}{\omega_{yL}y}$$

时可以对原函数 $f(x,y)$ 无失眞地复述，只要使采样间隔满足：

$$\varDelta x\leqslant\frac{\pi}{\omega_{xc}},\quad \varDelta y\leqslant\frac{\pi}{\omega_{yc}};\quad \omega_{xL}=\frac{2\pi}{\varDelta x},\quad \omega_{yL}=\frac{2\pi}{\varDelta y};$$

$\omega_{xL}>\omega_{xc}$，$\omega_{yL}>\omega_{yc}$，

这样离散化的数字图像就可以用内插函数 $g(x,y)$ 无失真地复原。但这里要求的内插函数 $\frac{\sin\omega_x x}{\omega_x x}$ 是一个有负值的函数，在具体用光学实现时做不到，例如CRT荧光屏上的一个光点或扫描器上的一个光点，其光强总是正的。但这样的一个光点其光强分布往往是从中心向外单调下降。因此如用一个三次 B 样条来表征这一光点是比较适宜的。在(7)式中就变成以 $B_3(x)\cdot B_3(y)$ 来代替 $g(x,y)$ 内插。当然这样一种替代将产生失真，失真量可以用样条函数理论来进行分析，即

$$\hat{f}(x,y)=\sum_m\sum_n f(m\Delta x,n\Delta y)B_3(x)B_3(y) \tag{8}$$

由于三次样条逼近是四阶微分逼近，如(4)式所示，其误差界依采样点间隔内 $f(x,y)$ 的四阶导数值的大小而定。即要看

$$\left\|\frac{\partial^4}{\partial x^4}f(x,y)\right\|_2,\ \left\|\frac{\partial^2}{\partial x^2}\ \frac{\partial^2}{\partial y^2}f(x,y)\right\|_2,\left\|\frac{\partial^4}{\partial y^4}f(x,y)\right\|_2$$

的大小，只要图像灰度在采样间隔内变化比较平滑，其四阶导数在间隔内的积分平均值较小，则内插误差也将很小。

样条函数的频谱是

$$F[B_k(x)]=\left(\frac{\sin\omega_x\frac{\Delta x}{2}}{\omega_x\frac{\Delta x}{2}}\right)^{k+1} \tag{9}$$

它基本上是低通型的传输函数，这种低通型的传输函数，对减少采样引起的混淆现象是有益的，因为混淆现象往往发生在频谱的高频区，对图像高频分量的衰减往往可以减轻混淆现象，当然这样对原图像的细节也将有所损失。

2）　图像数据压缩

在图像的数字传输过程中，往往需要压缩图像的巨大数据量，以减轻讯道的负担。一个经过采样的图像数据 $f(m\Delta x,n\Delta y)$ 有 $M\times N$ 个数据。这些数据中含有很大的讯息剩余量。因此，在进行图像数据的传输时为了得到更大的传输效率，要设法找出能代表图像主要讯息的特征量，这些量的数据量都应小于 $M\times N$。利用正交变换以减少图像数据的自由度(DOF)是行之有效的方法。SVD、K—L变换、富氏变换、Hadamard变换、余弦变换、Slant变换等都是行之有效的办法，但这些变换都有计算复杂、不利于计算机运算的缺点，尤其是对二維数据，数据量很大的情况下，更是如此。因此采用变换压缩的办法对图像传输的数据进行实时处理目前尚难具体实用。用 B 样条作为基函数对图像数据进行展开是解决计算量过大问题的一个途径。有可能解决图像的实时压缩问题。

设离散的采样图像数据为 $f(m\Delta x,n\Delta y)$，$m=1,\cdots,N,n=1,2,\cdots$；$N$ 为 $N\times N$ 个网格值。这些离散数据可以用 n_x 和 n_y 个 B 样条基函数来展开。对于二維图像数据可以展开为

$$\hat{f}(x,y)=\sum_{i=1}^{n_x}\sum_{j=1}^{n_y} b_{ij}B_3(x-x_i)B_3(y-y_j) \tag{10}$$

或写成矩阵表示形式

$$[\hat{f}(x_k,y_l)]=[B_3(x_k)][b_{ij}][B_3(y_l)]^T \tag{10$'$}$$

其结点为(x_i,y_j)；$i=1,2,\cdots,n_x$；$j=1,2,\cdots,n_y$，而各采样点已知值为$f(x_k,y_l)$；$k=1,2,\cdots,N$；$l=1,2,\cdots,N$．一般 $N>n_x$；$N>n_y$．而

$$[B_3(x_k)]=\begin{pmatrix} B_3(x_1-x_1') & B_3(x_1-x_2')\cdots\cdots & B_3(x_1-x_n') \\ B_3(x_2-x_1') & B_3(x_2-x_2') & \vdots \\ \vdots & & \vdots \\ B_3(x_N-x_1') & \cdots\cdots\cdots\cdots\cdots\cdots & B_3(x_N-x_n') \end{pmatrix}$$

为 $N\times n$ 矩阵，图象的 x 方向划分为 $x_1,x_2,\cdots,x_k,\cdots,x_N$ 个离散采样点，而 $x_1',x_2',\cdots,x_i',\cdots,x_n'$为样条函数的结点，都是均匀分布的，如图 4 所示。

图 4

$$[B_3(y_l)]=\begin{pmatrix} B_3(y_1-y_1') & B_3(y_1-y_2')\cdots\cdots & B_3(y_1-y_n') \\ B_3(y_2-y_1') & B_3(y_2-y_2')\cdots\cdots & \vdots \\ \vdots & & \vdots \\ B_3(y_N-y_1') & \cdots\cdots\cdots\cdots\cdots\cdots & B_3(y_N-y_n') \end{pmatrix}$$

$[b_{ij}]$ 是 $n\times n$ 系数矩阵，$[B_3(x_k)]$ 和 $[B_3(y_l)]$ 是由样条函数决定的限带状矩阵。

当结距为均匀时，$\rho=x_{i+1}-x_i=y_{j+1}-y_j$；$i,j=1,2,\cdots,n$时，因 $n\times n$ 个 B 样条函数对图像数据进行逼近时其最大误差限为

$$\|f-\hat{f}\|_2\leqslant 4\rho^4\pi^{-4}\left\{\left\|\frac{\partial^4}{\partial x^4}f(x,y)\right\|_2+\left\|\frac{\partial^2}{\partial x^2}\ \frac{\partial^2}{\partial y^2}f(x,y)\right\|_2+\left\|\frac{\partial^4}{\partial y^4}f(x,y)\right\|_2\right\}$$

在规定最大误差限时可以按 $f(x,y)$ 的导数的大小来决定 ρ 的大小，即所用的样条的数目的多少。在 n 数决定后，(10) 式中系数 $[b_{ij}]$ 的值可按最小二乘原则使下式最小化而求得：

$$\begin{aligned}\varepsilon&=\sum_{k,l}|f(x_k,y_l)-\hat{f}(x_k,y_l)|^2=\|[f(x_k,y_l)]-[\hat{f}(x_k,y_l)]\|_2^2\\&=\|[f(x_k,y_l)]-[B_3(x_k)][b_{ij}][B_3(y_l)]^T\|_2^2\end{aligned}$$

按 $\|A\|_2^2$ 可写成 $t_r(A^TA)$，上式可写成

$$\begin{aligned}\varepsilon&=t_r\{([f(x_k,y_l)]-[B_3(x_k)][b_{ij}][B_3(y_l)]^T)^T([f(x_k,y_l)]\\&\qquad-[B_3(x_k)][b_{ij}][B_3(y_l)]^T)\}\\&=t_r\{[f(x_k,y_l)]^T[f(x_k,y_l)]-([B_3(x_k)][b_{ij}][B_3(y_l)]^T)^T[f(x_k,y_l)]\\&\qquad-[f(x_k,y_l)]^T([B_3(x_k)][b_{ij}][B_3(y_l)]^T)\\&\qquad+([B_3(x_k)][b_{ij}][B_3(y_l)]^T)^T([B_3(x_k)][b_{ij}][B_3(y_l)]^T)\}\end{aligned}$$

求使 ε 为最小的 $[b_{ij}]$ 值，可对上式求导并使之等于零 $\frac{d\varepsilon}{d[b_{ij}]}=0$，得

$$\frac{d\varepsilon}{d[b_{ij}]}=0=-2[B_3(x_k)]^T[f(x_k,y_l)][B_3(y_l)]$$
$$+2[B_3(x_k)]^T[B_3(x_k)][b_{ij}][B_3(y_l)]^T[B_3(y_l)]$$

解之得

$$[b_{ij}]=\{[B_3(x_k)]^T[B_3(x_k)]\}^{-1}[B_3(x_k)]^T[f(x_k,y_l)][B_3(y_l)]\{[B_3(y_l)]^T[B_3(y_l)]\}^{-1} \tag{11}$$

(11)式中 $\{[B_3(x_k)]^T[B_3(x_k)]\}$ 是 $n\times n$ 方阵，一般讲是限带状矩阵，它的逆矩阵一般是存在的，并且不难计算。$[B_3(x_k)]$ 矩阵决定于 N/n 比，如 $\frac{N}{n}=2$ 则 $[B]$ 矩阵为七对角矩阵。条带数 $=4k-1$，$k=\frac{N}{n}$.

$$[B_3(x_k)]=\begin{pmatrix}
\frac{32}{48} & \frac{23}{48} & \frac{8}{48} & \frac{1}{48} & & & & \bigcirc \\
\frac{23}{48} & \frac{32}{48} & \frac{23}{48} & \frac{8}{48} & \frac{1}{48} & & & \\
\frac{8}{48} & \frac{23}{48} & \frac{32}{48} & & & \ddots & & \\
\frac{1}{48} & \frac{8}{48} & & \ddots & & & \ddots & \frac{1}{48} \\
 & \ddots & & & \ddots & & & \frac{8}{48} \\
 & & \ddots & & & \ddots & & \frac{23}{48} \\
\bigcirc & & & & \frac{1}{48} & \frac{8}{48} & \frac{23}{48} & \frac{32}{48}
\end{pmatrix} \text{ 为 } N\times n \text{ 矩阵.}$$

但求逆矩阵运算总是比较繁复的，不容易做到实时处理。例如要做到图像传输的速度（从几秒一帧到每秒几十帧）按陆地卫星照片的 2340×3240 像素的数据量，就比较难以达到，下面利用样条函数的特性引出一个简化的压缩方案。

当 B 样条函数以最小二乘原则逼近图象函数 f 时其误差极限为

$$\|f-\hat{f}\|_2\leqslant4\pi^{-4}h^4\|D^4f\|_2$$

而 $\hat{f}$ 的高阶导数也以最小二乘逼近函数 f 的相应阶导数，对于 $2m-1$ 次逐段多项式

$$\|D^j(f-\hat{f})\|_2\leqslant\pi^{2m-j}h^{2m-j}\|D^{2m}f\|_2$$

利用B样条的导数特性

$$B'_k(x)=B_{k-1}\left(x+\frac{1}{2}\right)-B_{k-1}\left(x-\frac{1}{2}\right)$$

因此可得

$$D^3\hat{f}(x_k,y_l)=\sum_{i=1}^{n}\sum_{j=1}^{n}C_{ij}B_0(x-x_i)B_0(y-y_j) \tag{12}$$

(12)式中 $B_0(x)$ 为零次样条函数，它是一单位脉冲函数。C_{ij} 是系数 b_{ij} 的三次差分，即 $[C_{ij}]=\nabla^3[b_{ij}](\nabla^3)^T$.

$$B_0(x)=\begin{cases}1 & x\in(x_i,x_{i+1})\\ 0 & 其它\end{cases}$$

用这样一个函数的集合去逼近 D^3f，很显然系数 C_{ij} 应等于间隔(x_i,x_{i+1}) 和 (y_j,y_{j+1}) 小方块内D^3f的平均值，即

$$C_{ij}=\frac{1}{(x_{i+1}-x_i)(y_{j+1}-y_j)}\int_{x_i}^{x_{i+1}}\int_{y_j}^{y_{j+1}}D_x^3D_y^3f(x,y)dxdy \tag{13}$$

令 $\rho_x=x_{i+1}-x_i$，$\rho_y=y_{j+1}-y_j$，则得到的估值为

$$D_x^3D_y^3\hat{f}(x,y)=\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{1}{\rho_x\rho_y}\int_{x_i}^{x_{i+1}}\int_{y_j}^{y_{j+1}}D_x^3D_y^3f(\alpha,\beta)d\alpha d\beta B_0(x-x_i)B_0(y-y_j)$$

而被逼近的图像的导函数 $D_x^3D_y^3f(x_k,x_l)$ 则可从离散图像 $f(x_k,y_l)$ 的矩阵 $[F]$ 通过差分运算近似获得，即求运算

$$\nabla^3F(\nabla^3)^T;\quad \nabla=\frac{1}{\rho}\begin{pmatrix}-1 & 1 & & & & \bigcirc\\ 0 & -1 & 1 & & & \\ 0 & 0 & -1 & 1 & & \\ & & & \ddots & \ddots & \\ & \bigcirc & & -1 & 1 & 0\\ & & & & -1 & 1\end{pmatrix}$$

ρ为结距 $\rho=\rho_x=\rho_y$. F和∇都是$N\times N$ 矩阵，因此，在离散图像的情况下，C_{ij} 是在结距范围内离散图象灰度的三次差分的相加平均值。用这种方法对图像数据进行压缩后只要传输系数 C_{ij}. 在接收端收到系数 C_{ij} 后，再进行积分运算以得到原图像，但积分运算必须有边界条件来得出积分常数，在这里边界条件是$\hat{f}(x_k,-1)$ 和$\hat{f}(-1,y_l)$，即图像边界处的灰度值，这些边界处像素的灰度值必须作为额外讯息传输至接收端。积分运算用数字的方法进行要比差分复杂，例如可用改进的梯形公式进行积分近似，最后再加入边界值。这种发端的差分求和运算和收端的数字积分运算可以做成硬件形式。发端的差分求和运算设备比较简单，而收端则较为复杂。这比较适合于陆地卫星上图像数据压缩的情况，因为收端处理可以在地面站慢慢地进行，可以做比较复杂的运算。图 5 是这种压缩传输的设备框图。全部由数字设备组成。

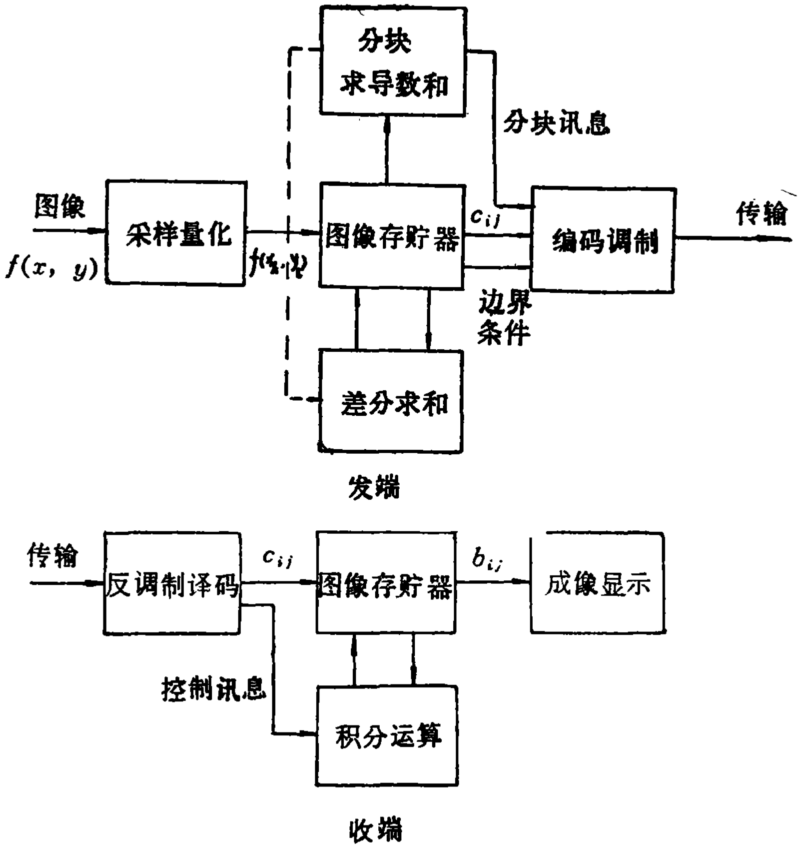

图　5

这种压缩方法的另一个问题是结距选择问题，用样条函数逼近一幅图像时其误差极限是与结距ρ和图像在该处的四阶导数的乘积有关的，如(4)式所示。按(4)式，在图像变化比较急剧的地方，结距ρ应取得小些，因此结距应该是各处不同的，但不均匀结距会给计算上带来一些麻烦。在图像压缩中可采用分块等结距的办法，例如图6为一幅256×256像素的数字化图像，可分成8×8＝64小块，各区的结距不相等，从6×6到32×32不等。这样做法可以做到在一定误差极限下达到较大的压缩比。

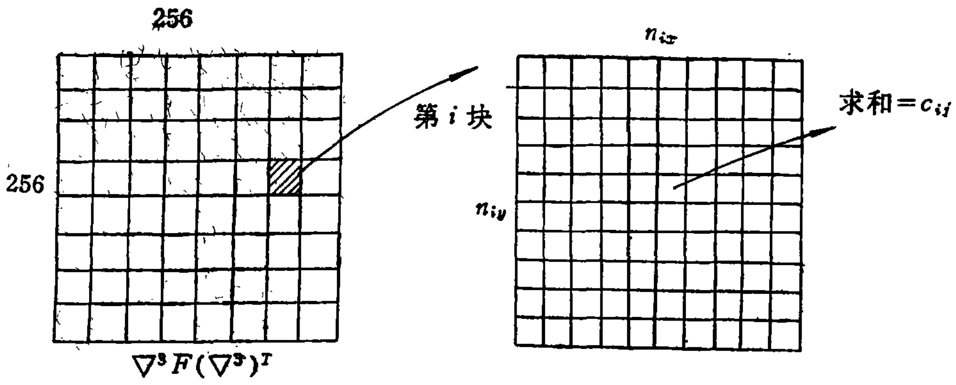

图　6

用样条函数对图像函数进行压缩，在概念上与富氏变换、K—L变换、Hadamard 变换等变换压缩有所不同，变换压缩是把图像数据先变换到讯息量更为集中的向量空间，然后对变换后的数据进行截取，使数据量压缩。而样条函数逼近是根据某个区域中图像导数（差分近似）的平均值来决定所用样条基函数的数目（即结距），而用一定数目的基函数的系数来表示图像特征的，因此是先截取后展开的形式。由于这种方式上的改变，使得展开运算中的維数减少，因而从根本上简化了计算。这种概念也可以用在其它数据压缩的場合。

3） 图像数据平滑

当我们对数据值知道得不确切，即存在着干扰时，样条函数也可用来作平滑处理，对随机干扰进行平滑，这也可称为样条图像修复。一个线性的位移不变的成像系统可以写成卷积形式

$$g(x,y)=\iint h(x-\xi,y-\eta)f(\xi,\eta)d\xi d\eta+n(x,y)$$

进行样条函数平滑的第一步是把原图像和点散布函数 $h(x,y)$ 都用 B 样条基函数展开

$$f(x,y)=\sum_{i=1}^{n}\sum_{j=1}^{n}b_{ij}B_k(x-x_i)B_k(y-y_j)$$

$$h(x,y)=\sum_{p=1}^{n}\sum_{q=1}^{n}d_{pq}B_l(x-x_p)B_l(y-y_q)$$

$B_k(x)$ 和 $B_l(x)$ 分别为 k 次和 l 次样条函数，则

$$\begin{aligned}g(x,y)=&\sum_p\sum_q\sum_i\sum_j b_{ij}d_{pq}\iint B_k(\xi-x_i)B_k(\eta-y_j)B_l(x-\xi-x_p)B_l\\&(y-\eta-y_q)d\xi d\eta+n(x,y)\\=&\sum_p\sum_q\sum_i\sum_j b_{ij}d_{pq}B_k(x-x_i)\circledast B_l(x-x_p)\cdot B_k(y-y_j)\circledast B_l(y-y_q)+n(x,y)\end{aligned} \tag{14}$$

由于 $B_k(x-x_i)\circledast B_l(x-x_p)=B_{k+l+1}(x-x_i-x_p)$，再假设每样条函数的结距取得一样，即 $\rho=x_{i+1}-x_i=x_{p+1}-x_p=y_{j+1}-y_j=y_{q+1}-y_q$ ，则可写成

$$g(x,y)=\sum_i\sum_j g_{ij}B_{k+l+1}(x-i\rho)B_{k+l+1}(y-j\rho)+n(x,y) \tag{15}$$

利用 (14) 式和 (15) 式可写成系数的矢量表示式

$$\boldsymbol{g}=[H]\boldsymbol{f}+\boldsymbol{n}$$

式中 $\boldsymbol{g}$ 是系数 g_{ij} 形成的链串列矢量，維数为 N^2, $\boldsymbol{f}$ 是图像的样条函数展开中的系数 b_{ij} 的 N^2 維列矢量，$[H]$ 是成像系统的系数 d_{pq} 组成的 $N^2\times N^2$ 变换矩阵（这一矩阵的形成方法在这里不介绍了，可参看 Pratt 的书）。当成像系统是可分离时，上式可写成矩阵形式

$$[G]=[H_x][F][H_y]+[n_{xy}] \tag{16}$$

式中 $[H]=[H_x]\otimes[H_y]$, $\otimes$表示矩阵的直积。

上式中各元已不是像素而是等节距的样条函数的系数，維数比像素的維数小得多。从(15)式可以看到，用样条函数对图像和点散布函数进行逼近后，系统输出函数的次数是原图像展开所用样条函数的次数和点散布函数所用样条函数的次数之和。这就非常形象地表示出输出函数由于点散布函数的作用比输入函数更为平缓，形象地表示出系统的模

糊作用。也可以说样条函数是非常适合于这种模糊系统的描述的。

现在平滑的问题变为已知 $[g_{ij}]$ 和 $[H_x]$、$[H_y]$ 矩阵并已知 $[n_{xy}]$ 的统计特性的情况下求最佳的系数矩阵 $[b_{ij}]$ 的问题。下面先就无系统模糊的情况下讨论对随机干扰进行平滑的方法。以连续形式表示为

$$g(x,y)=f(x,y)+n(x,y)$$

用最小二乘进行平滑，要求在约束条件

$$\sum_i\sum_j\left(\frac{f(x_i,y_j)-g(x_i,y_j)}{\delta g(x_i,y_j)}\right)^2\leqslant S$$

的情况下，使 $\iint f''(x,y)^2dxdy$ 最小化。其中 S 为平滑程度的指标，与 $\delta g(x_i,y_j)$ 有关。$\delta g(x_i,y_j)$ 是干扰引起的测量误差量。当 $S=0$ 时就变成内插问题，$g(x_i,y_j)$ 为测量到的像素网格上的値，可写成 g_{ij}. $f(x,y)$ 为被平滑的函数，用变分法解，可使下列函数最小化，

$$\iint f''(x,y)^2dxdy+p\left\{\sum_i\sum_j\left(\frac{f(x_i,y_j)-g_{ij}}{\delta g_{ij}}\right)^2+z^2-S^2\right\}$$

假设干扰是对各像素独立的均匀白噪声引起的，可把上式写成离散形式，如下式用矩阵表示为：

$$\|[Q][F][Q]^T\|_2^2+\frac{p}{\sigma^2}\{\|[F]-[G]\|_2^2+z^2-S^2\}=W(F) \qquad (17)$$

上式中 $[Q]$ 为二阶差分算子

$$[Q]=\begin{bmatrix} -2 & 1 & 0 & & & \\ 1 & -2 & 1 & & \bigcirc & \\ 0 & 1 & -2 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & \bigcirc & & 1 & -2 & 1 \\ & & & & 1 & -2 \end{bmatrix}$$

我们使 $f(x,y)$ 用三次 B 样条函数展开为

$$[F]=[T][b_{ij}][T]^T \qquad (18)$$

这里

$$[F]=\begin{bmatrix} f_{11} & f_{12} & \cdots & f_{1N} \\ f_{21} & & & \vdots \\ \vdots & & & \vdots \\ f_{N1} & & \cdots\cdots & f_{NN} \end{bmatrix}$$

并假设样条函数的结距即为采样间隔，相当于2)节中 $\frac{N}{n}=1$ 的情况，这时

$$[T]=\begin{bmatrix} \frac{4}{6} & \frac{1}{6} & & & \bigcirc \\ \frac{1}{6} & \frac{4}{6} & \frac{1}{6} & & \\ & \ddots & \ddots & \ddots & \\ & & & & \frac{1}{6} \\ \bigcirc & & & \frac{1}{6} & \frac{4}{6} \end{bmatrix}$$

为三对角矩阵，并为 $N\times N$ 方阵。当然这里也可取比较少的样条函数来逼近，并容许一定的误差，这时 $[T]$ 矩阵就像前节所示变为更多对角项的矩阵。而且将成为 $N\times n$ 矩阵，不再是方阵。按这样的一些符号关系 (18) 式变为

$$W(b_{ij})=\|[Q][T][b_{ij}][T]^T[Q]^T\|_2^2+\frac{p}{\sigma^2}\{\|[T][b_{ij}][T]^T-[G]\|_2^2+z^2-S^2\}$$

对上式求极值，即使 $\frac{dW([b_{ij}])}{d[b_{ij}]}=0$，得

$$\frac{\sigma^2}{p}[Q]^T[Q][T][b_{ij}][T]^T[Q]^T[Q]+[T][b_{ij}][T]^T=[G]$$

令 $\lambda=-\frac{1}{p}$，则上式可化简为

$$\left(I+\frac{\sigma}{\sqrt{p}}[Q]^T[Q]\right)[T][b_{ij}][T]^T\left(I-\frac{\sigma}{\sqrt{p}}[Q]^T[Q]\right)=[G]$$

可解出

$$[b_{ij}]=[T]^{-1}(I+\sqrt{\lambda}\sigma[Q]^T[Q])^{-1}[G](I-\sqrt{\lambda}\sigma[Q]^T[Q])^{-1}([T]^T)^{-1} \tag{19}$$

(19) 式中 $[Q]^T[Q]$ 为对称矩阵，因此括号中矩阵的逆阵总是存在的，并在计算上比较方便，当 $\frac{N}{n}>1$ 时 $[T]$ 将为长方矩阵，长方矩阵的逆矩阵不存在，只能求其伪逆。(19)式中 p 为拉格朗日常数，它是用来表示平滑的程度的，p 小则平滑程度增加，但对图像中高频分量有损失，p 大则平滑程度减弱。如要用 (19) 式的简化公式来计算则 p 应为负数，否则 $\sqrt{\lambda}$ 将为虚数，得出的 $[b_{ij}]$ 有复数值，将使计算复杂化，而且由于图像灰度无法用复数表示而产生失真。

当系统有模糊作用时，系统方程为

$$[G]=[H_x][F][H_y]+[n] \tag{20}$$

用同样的方法在限制条件

$$\left\| \frac{1}{\sigma^2}([H_x][F][H_y]-[G]) \right\|_2^2 \leqslant S$$

下求$\iint f''(x,y)^2dxdy$ 最小化，在 $[H_x]=[H_y]$ 即模糊函数在 x 和 y 方向一致时可写成

$$[b_{ij}]=[T]^{-1}([H]^T[H]+\sqrt{\lambda}\sigma[Q]^T[Q])^{-1}[G]([H]^T[H] - \sqrt{\lambda}\sigma[Q]^T[Q])^{-1}([T]^T)^{-1} \tag{21}$$

用样条函数平滑的效果可以和Wiener 滤波相比拟，只是 Wiener 滤波中的图像协方差矩阵 $[C_f]$ 在这里给算子 $[Q]$ 和 $[T]$ 的组合所代替。这是因为样条函数只是从灰度变化的光滑性上进行逼近，并使误差最小，而并沒有考虑到图像的统计特性的缘故。另外样条函数本身带来了计算上的简便，使平滑的计算简单化。

三、结　语

样条函数对图像函数具有良好的逼近性能，尤其是对那些经过成像系统模糊的图像，因为这样的图像本身就具有脉冲模糊性能。用样条函数能对采样、成像、显示系统作很好的模拟，既利于掌握图像系统的实质，也得到了计算上的方便。在图像数据简缩中样条函数具有先简缩后展开的特点，使得逼近展开时維数降低，大为减轻了二維正交展开中计算的繁复性。因此样条函数应该在数字图像的计算机处理上具有良好的前景，但在理论上和方法上现在还刚开始，在样条展开中不涉及图像的统计特性，这与一般的压缩和滤波方法有很大的不同，但其效果和应用范围究竟怎样，还有待进一步研究。本文只是提出一些粗浅的设想，希望引起有关这方面工作的同志的讨论和批评。

参考文献

[1] M.H.Schultz, “样条分析”，科学出版社，1973.

[2] 李岳山、齐东旭，“样条函数方法”，科学出版社，1979.

[3] D. G. McCaughey, “Variable Knot Splines as Images Analysis Technique”, Image Science, Math 1976, No.V, P. 168—175.

[4] D. G. McCaughey, “An Image Coding Algorithm Using Spline Functions”, SPIE, Vol. 149, Applications of Digital Image Processing, 1978, P.51—61.

[5] M. J. Peyrovian and A.A.Sawchuk, “Image Processing by Smoothing Spline Function”, SPOIE, Vol.74, Image Processing, 1976, P. 25—30.

[6] W. K. Pratt, “Digital Image Processing”, John Wiley & Sons, 1978.

The Applications of the Spline Function in the Image Processing and Transmission

Wang Shao—ling

Abstract

Spline function is available to the numerical computation on the digital computer for its ease to process. Spline function has been utilized in all of the engineering territories. In the discrete digital processing of the continuous images, there are very large amount of image data. If we use the spline function to interpolate and smooth the data set, we would got advantages in computation time reduction. In this paper, we try to discuss an efficient image data compressing method using spline function for least hardware implementations; and a restoration procedure to smooth out the random noise distorted image with the spline functions.