

# 基于神经网络的 NURBS 曲面重建

肖宜龙, 路游, 亢永刚

(中国石油大学 计算机科学与技术系, 北京 102249)

**摘要:** 曲面重建是 CAGD 中的重点研究课题, 而神经网络具有很好的非线性逼近能力, 文中将二者结合, 给出了一种利用神经网络将三维数据点拟合为 NURBS 曲面的方法。提出的前馈型神经网络包含四个隐层, 其中一层的激活函数为 B 样条基函数。由数学推导可知, 该网络可以表达 NURBS 曲面, 通过对控制顶点及其权重的学习, 可以用该网络来重建 NURBS 曲面。权值的调整通过误差反传与梯度下降法实现。实验结果表明, 文中提出的方法是可行的。

**关键词:** 曲面重建; 神经网络; NURBS 曲面

**中图分类号:** TP311

**文献标识码:** A

**文章编号:** 1673-629X(2009)09-0065-04

## NURBS Surface Reconstruction Based on Neural Network

XIAO Yi-long, LU You, QI Yong-gang

(Department of Computer Science and Technology, China University of Petroleum, Beijing 102249, China)

**Abstract:** Surface reconstruction is the focus in CAGD research. Neural network has a good ability of non-linear approximation. In this paper, give a method for the NURBS surface fitting using neural network. The proposed feed-forward neural network contains four hidden layers. The activation function of one layer is the B-spline basic function. From mathematical derivation know that NURBS surface can be expressed by this network. Through the study of the control points and weights, the network can be used to NURBS surface reconstruction. Experiments show that the method is feasible.

**Key words:** surface reconstruction; neural networks; NURBS surface

## 0 引言

利用三维扫描仪从物体表面取得散乱数据, 并对其曲面重建是 CAGD 中的热点研究课题。在涉及自由曲面的工业造型领域有广泛的应用。曲面重建的方法通常有: 曲面拟合法<sup>[1~3]</sup>, 这是一类基于参数曲面、隐函数曲面或分段多项式近似数据点的方法, 它主要包括隐式拟合和显式拟合两种方法; 分段线性重建法<sup>[4]</sup>, 这类方法是通过建立多面体化的表面, 插值或近似给定点, 该类方法又可分为多边形方法和体方法; 基于物理的重建法<sup>[4]</sup>, 这是一种对于构造的初始几何模型, 通过引入物体本身的物理特征和所处外部环境因素的描述使模型变形并逼近散乱点的方法; 基于神经网络的重建法<sup>[5,6]</sup>等等。但是在曲面重建领域没有一种方法是十全十美的。考虑到神经网络强大的泛函逼近能力、容错性能以及联想力, 结合曲面造型方法中常

用的 NURBS 技术, 提出了一种基于前馈神经网络的神经网络 NURBS 曲面重建方法。

## 1 NURBS 曲面简介

张量积双参数 NURBS 曲面的定义公式如下<sup>[7]</sup>:

$$S(u, v) = \frac{\sum_{k=0}^n \sum_{r=0}^m N_{k,p}(u) N_{r,q}(v) W_{k,r} P_{k,r}}{\sum_{k=0}^n \sum_{r=0}^m N_{k,p}(u) N_{r,q}(v) W_{k,r}} \quad (1)$$

其中  $S(u, v)$  是 NURBS 曲面上参数为  $(u, v)$  的一点,  $N_{r,q}(v)$  和  $N_{k,p}(u)$  是次数分别为  $p$  和  $q$  的 B 样条基函数,  $P_{k,r}$  是曲面的控制网格上的控制顶点,  $W_{k,r}$  是控制顶点  $P_{k,r}$  的权重,  $W_{k,r}$  越大, 曲面就越靠近控制点  $P_{k,r}$ 。  $U = [u_0, \dots, u_{k+p+1}]$  和  $V = [v_0, \dots, v_{r+q+1}]$  是节点向量。

## 2 神经网络

1962 年, Rosenblatt 给出了人工神经网络著名的学习定理: 人工神经网络可以学会它可以表达的任何东西。这一部分将给出一个神经网络<sup>[8]</sup>, 通过数学推导可以证明, 这一网络可以表达 NURBS 曲面。

收稿日期: 2008-12-05; 修回日期: 2009-03-23

基金项目: 国家自然科学基金(60873093)

作者简介: 肖宜龙(1983-), 男, 山西朔州人, 硕士研究生, 研究方向为计算机辅助几何设计; 路游, 副教授, 研究方向为计算机辅助几何设计。

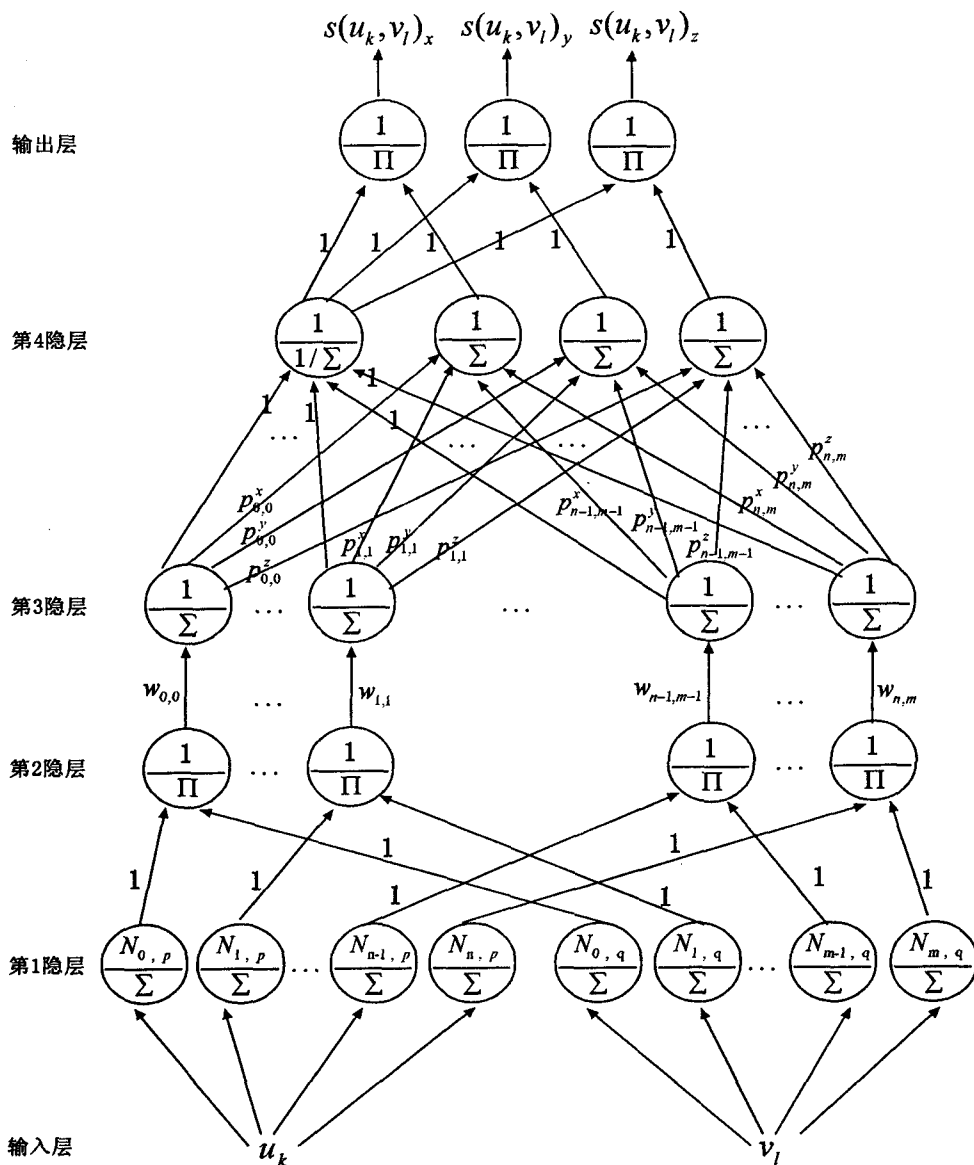


图 1 神经网络结构图

图 1 表示的是包含四个隐层的前馈神经网络。第一隐层的激活函数为 B 样条基函数,第二隐层、第三隐层、第四隐层及输出层的激活函数为斜率为 1 的线性函数。第一隐层及第三隐层的输入累积函数为线性求和函数;第二隐层及输出层的输入累积函数为多项式函数(求积);第四隐层有四个神经元,右边三个神经元的输入累积函数为线性求和函数,左边一个神经元的输入累积函数为反商函数(先对输入求和,再对和求倒数)。网络中需要学习调整的向量为控制顶点  $P_{k,r}$  及其权重  $w_{k,r}$ 。

用  $\text{net } r I(\cdot)$  和  $\text{net } r O(\cdot)$  分别表示第  $r$  隐层神经元的输入和输出。按照图 1,第 1 隐层的输出为:

$$\text{net } 1 O(i) = N_{i,p}(u_k) \quad (2)$$

$$\text{net } 1 O(j) = N_{j,q}(v_l) \quad (3)$$

因为第 2 隐层、第 3 隐层及第 4 隐层的激活函数都

是斜率为 1 的线性的函数,所以有:

$$\text{net } 2 O(i,j) = \text{net } 2$$

$$I(i,j) = \text{net } 1 O(i) \cdot \text{net } 1 O(j) \quad (4)$$

$$\text{net } 3 O(i,j) = \text{net } 3 I(i,j) = w_{i,j} \cdot \text{net } 2 O(i,j) \quad (5)$$

$$\text{net } 4 O(L) = \text{net } 4 I(L) =$$

$$\frac{1}{\sum \sum \text{net } 3 O(i,j)} \quad (6)$$

$$\text{net } 4 O(R) = \text{net } 4 I(R) = \sum \sum p_{i,j} \text{net } 3 O(i,j) \quad (7)$$

其中  $\text{net } 4 I(L)$  和  $\text{net } 4 O(L)$  分别表示第四隐层左边一个神经元的输入和输出,  $\text{net } 4 I(R)$  和  $\text{net } 4 O(R)$  分别表示第四隐层右边三个神经元的输入和输出。另外,输出层的累积输入函数为乘积函数,激活函数的斜率为 1 的线性函数,因此

$$S(u_k, v_l) = \text{net } 4 O(L) \cdot \text{net } 4 O(R) =$$

$$\frac{\sum \sum p_{i,j} \text{net } 3 O(i,j)}{\sum \sum \text{net } 3 O(i,j)} \quad (8)$$

把式(2) ~ (5) 代入式(8),得到

$$S(u_k, v_l) = \text{net } 4 O(L) \cdot \text{net } 4 O(R) =$$

$$\frac{\sum \sum w_{i,j} N_{i,p}(u_k) N_{j,q}(v_l) p_{i,j}}{\sum \sum w_{i,j} N_{i,p}(u_k) N_{j,q}(v_l)} \quad (9)$$

其中  $S(u_k, v_l)$  表示  $(S(u_k, v_l)_x, S(u_k, v_l)_y, S(u_k, v_l)_z)$ ,  $P_{i,j}$  表示  $(p_{i,j}^x, p_{i,j}^y, p_{i,j}^z)$ , 式(9)即为 NURBS 曲面的表达式。由此可知,本节所构建的神经网络可以表达 NURBS 曲面,故通过对网络中控制顶点  $P_{k,r}$  及其权重  $w_{i,j}$  的学习,可以用该网络将原始数据点拟合为 NURBS 曲面。

## 2.1 控制顶点的学习规则

定义网络的误差测度为:

$$E(p, w) = \frac{1}{2} (I_{k,l} - S(u_k, v_l))^2$$

其中  $I_{k,l}$  是对应于某一参数对  $(u_k, v_l)$  的期望输出,  $S(u_k, v_l)$  是把  $(u_k, v_l)$  输入网络后的实际输出。按照负梯度下降法,为了使  $E$  达到极小点,应该有

$$\Delta p_{i,j} \propto -\frac{\partial E}{\partial p_{i,j}}$$

$$\text{因为 } \frac{\partial E}{\partial p_{i,j}} = \frac{\partial E(p, w)}{\partial S(u_k, v_l)} \cdot \frac{\partial S(u_k, v_l)}{\partial p_{i,j}} = (S(u_k, v_l) - I_{k,l}) \cdot \frac{\text{net3o}(i, j)}{\sum_{i=0}^n \sum_{j=0}^m \text{net3o}(i, j)}$$

$$\text{故 } \Delta p_{i,j} = -\eta \frac{\partial E}{\partial p_{i,j}} = -\eta (S(u_k, v_l) - I_{k,l}) \cdot \frac{\text{net3o}(i, j)}{\sum_{i=0}^n \sum_{j=0}^m \text{net3o}(i, j)} \quad (10)$$

(10) 式即为  $p_{k,r}$  的调整规则,  $\eta$  为学习率。

## 2.2 权重的学习规则

定义网络的误差测度为:

$$E(p, w) = \frac{1}{2} (I_{k,l} - S(u_k, v_l))^2$$

按照负梯度下降法,为了使  $E$  达到极小点,应该有

$$\Delta w_{i,j} \propto -\frac{\partial E}{\partial w_{i,j}}$$

因为

$$\begin{aligned} \frac{\partial E}{\partial w_{i,j}} &= \frac{\partial E(p, w)}{\partial S(u_k, v_l)} \cdot \frac{\partial S(u_k, v_l)}{\partial \text{net3o}(i, j)} \cdot \frac{\partial \text{net3o}(i, j)}{\partial w_{i,j}} \\ &= (S(u_k, v_l) - I_{k,l}) \cdot \frac{p_{i,j} \sum \sum \text{net3o}(i, j) - \sum \sum p_{i,j} \text{net3o}(i, j)}{(\sum \sum \text{net3o}(i, j))^2} \\ &\quad \cdot \text{net2o}(i, j) \end{aligned}$$

故

$$\begin{aligned} \Delta w_{i,j} &= -\lambda \cdot (S(u_k, v_l) - I_{k,l}) \cdot \frac{p_{i,j} \sum \sum \text{net3o}(i, j) - \sum \sum p_{i,j} \text{net3o}(i, j)}{(\sum \sum \text{net3o}(i, j))^2} \\ &\quad \cdot \text{net2o}(i, j) \end{aligned} \quad (11)$$

(11) 式即为  $w_{i,j}$  的调整规则,  $\lambda$  为学习率。

## 3 NURBS 曲面重建方法

上一部分给出使用的神经网络的结构及其各层神经元权向量的调整公式;这一部分将给出利用这一神经网络重建曲面的方法。

图1表示的神经网络,输入为参数  $(u_k, v_l)$ , 输出为  $(S(u_k, v_l)_x, S(u_k, v_l)_y, S(u_k, v_l)_z)$  即空间中的一个点。通常要对空间数据点进行 NURBS 重建, 已知的条件为这些点的空间坐标  $(x, y, z)$ , 因此这些点是无法直接输入图1表示的神经网络进行训练的, 这就需

要先把这些点参数化, 得到每个点对应的参数  $(u, v)$ , 然后形成样本集  $((u, v), (x, y, z))$ , 再把这个参数化后的样本集输入神经网络进行训练。考虑到本文所使用的数据点参数化方法的特点, 假设所给的空间数据点集是矩形排列的。

### 3.1 数据点的参数化

假设矩形排列的数据点集为  $\{p_{i,j}\} i=1, 2, \dots, n; j=1, 2, \dots, m$ , 采用弦长参数化方法对数据点集参数化, 并确定曲面的节点向量。

$u$  向第  $i$  行  $(i=1, 2, \dots, n)$  数据点  $p_{i,j}, j=1, 2, \dots, m$  的参数化方法为:

$$\begin{aligned} \text{令 } d &= \sum_{k=2}^m \|p_{i,k} - p_{i,k-1}\| \\ \text{则 } \bar{u}_{i,1} &= 0, \bar{u}_{i,m} = 1 \\ \bar{u}_{i,k} &= \bar{u}_{i,k-1} + \frac{\|p_{i,k} - p_{i,k-1}\|}{d}, k=2, \dots, m-1. \end{aligned}$$

$v$  向第  $j$  列  $(j=1, 2, \dots, m)$  数据点  $p_{i,j}, i=1, 2, \dots, n$ , 的参数化方法类似, 不再叙述。

公共的  $u$  向参数值可取为  $u'_j = \frac{1}{n} \sum_{i=1}^n \bar{u}_{i,j}, j=1, 2, \dots, m$ 。公共的  $v$  向参数化方法类似。 $u$  向和  $v$  向的参数化完成以后, 每个数据点  $p_{i,j}$  就对应了一个参数值  $(\bar{u}_{i,j}, \bar{v}_{i,j})$ 。

$u$  向的节点向量定义为:

$$u_0 = \dots = u_p = 0, u_m = \dots = u_{m+p} = 1$$

$$u_{j+p} = \frac{1}{p} \sum_{i=j}^{j+p-1} u'_{i+1} \quad j=1, \dots, m-p-1$$

其中  $p$  为 B 样条基函数的次数。 $v$  向节点向量的定义方法类似。

### 3.2 神经网络的训练算法

第三部分给出的  $P_{i,j}$  及  $W_{i,j}$  的调整公式是依据误差反传原理得到的, 这与 BP 网络中权值的调整规则是一致的, 因此参考已有的 BP 学习算法<sup>[9]</sup>, 文中使用的神经网络训练算法流程设计为<sup>[9,10]</sup>:

(1) 初始化。控制顶点的初始值赋随机数, 权重的初始值赋为 1;

(2) 依次输入参数化后的训练样本对, 计算输出;

(3) 计算每个样本的网络输出误差及网络累积总误差;

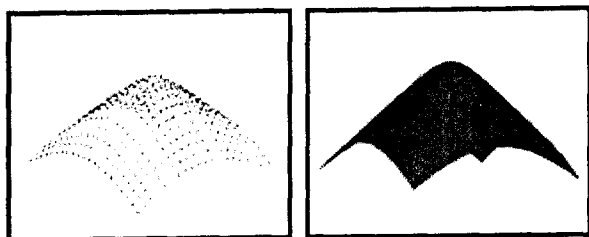
(4) 按公式(10)、(11) 调整隐层中各  $P_{i,j}$  及  $W_{i,j}$  的值;

(5) 检查是否所有训练样本完成一轮训练, 若没有, 返回步骤(2), 否则转步骤(6);

(6) 检查网络总误差是否达到给定的精度要求, 若是, 训练结束, 若不是返回步骤(2)。

## 4 仿真结果

图 2 表示用文中方法做仿真实验得到的结果。(a)表示空间中矩形排列的原始数据,共有 441 个三维数据点,是从解析方程  $z = \sqrt{3 \times (1 + x^2/4 + y^2/4)}$  采样得到的。(b)是用文中方法得到的双三次 NURBS 曲面的光照图。



(a) 原始数据点

(b) 拟合后的 NURBS 曲面

图 2 实验仿真

## 5 结束语

曲面重建技术是逆向工程中的热点研究课题,广泛应用于航空、汽车、船舶以及其它工业造型领域。NURBS 技术作为国际标准,具有统一、通用、有效的标准算法和相应的配套技术,可以表示自由曲线曲面,因此在重构技术中被广泛采用。而神经网络具有强大的非线性学习能力和一定的抗噪性。文中正是考虑到这些特点提出了一种基于神经网络的 NURBS 曲面重建方法,把参数化后的数据交由神经网络去学习进而拟合出相应的曲面。

仿真实验表明该方法是可行的,对原始数据的拟

合度较高。然而该方法处理的是规则的矩形排列的数据点集,因此在下一步的研究中将考虑大规模散乱数据点集的压缩和神经网络曲面重建。同时神经网络的训练算法也会进一步改进。

### 参考文献:

- [1] Piegl L, Tiller W. The NURBS Book[M]. Berlin: Springer Verlag, 1997: 353-363.
- [2] 祁伟丽, 秦新强. 基于 B 样条的平面轮廓重构闭合曲面算法[J]. 计算机技术与发展, 2008, 18(7): 112-115.
- [3] Ma Weiyin, Kruth J P. Parameterization of randomly measured points for least squares fitting of B-spline curves and surfaces[J]. Computer Aided Design, 1995(9): 663-675.
- [4] 武剑洁, 王启付, 黄运保, 等. 逆向工程中曲面重建的研究进展[J]. 工程图学报, 2004(2): 133-142.
- [5] Barhak J, Fischer A. Adaptive reconstruction of freeform objects with 3D SOM neural network grids[J]. Computers & Graphics, 2002(26): 745-751.
- [6] 周金字, 谢里阳. 基于神经网络预拟合的 B 样条曲面反求[J]. 东北大学学报: 自然科学版, 2003(6): 556-559.
- [7] 杨广全, 李思益, 曹西京. CAGD 中的 NURBS 曲面——理论与应用[J]. 西北轻工业学院学报, 2001(3): 62-65.
- [8] Cheng Ming-Yang, Wu Hung-Wen, Su A Wen-Yu. On Non-Uniform Rational B-spline Surface Neural Networks[J]. Neural Process Lett, 2008(28): 1-15.
- [9] 韩力群. 人工神经网络理论、设计及应用[M]. 北京: 化学工业出版社, 2005: 43-49.
- [10] 杜华英, 赵跃龙. 人工神经网络典型模型的比较研究[J]. 计算机技术与发展, 2006, 16(5): 97-99.

(上接第 64 页)

## 4 结束语

文中将主成分分析、遗传算法融合在 BP 神经网络中, 先利用主成分分析进行样本降维, 再利用遗传算法优化初始权值, 最后进行 BP 神经网络训练, 形成一种提高 BP 神经网络泛化能力的改进算法, 为实际生活中样本复杂的神经网络提出了新的思路。通过对输入样本集进行主成分分析处理, 达到了数据降维、剔除冗余信息的目的, 借助遗传算法能够全局寻优的特点对初始权值进行优化, 弥补了标准 BP 算法容易陷入到局部极值中的不足, 增加了网络对训练集合以外样本的识别能力, 提高了 BP 神经网络的泛化能力。

### 参考文献:

- [1] Lehtokangas M. Modified Cascade - Correlation Learning for Classification[J]. IEEE Trans. on Neural Networks, 2000, 11(3): 795-798.
- [2] 武 妍, 张立明. 神经网络的泛化能力与结构优化算法研究[J]. 计算机应用研究, 2002(6): 21-25.
- [3] 周开利, 康耀红. 神经网络模型及其 MATLAB 仿真程序设计[M]. 北京: 清华大学出版社, 2006: 69-100.
- [4] 杨 静, 毛宗源. 基于 PCA 和神经网络的识别方法研究[J]. 计算机工程与应用, 2007, 43(25): 246-248.
- [5] Ruan Qing, Wang Yi-qiang. PCA Approach to BP Learning[J]. Journal of Fudan University: Natural Science, 2005, 44(2): 318-322.
- [6] 王智平, 刘在德, 高成秀, 等. 遗传算法在 BP 网络权值学习中的应用[J]. 甘肃工业大学学报, 2001, 27(2): 20-22.
- [7] Zorriassatine F, Tannock J D T. A review of neural networks for statistical process control[J]. Journal of Intelligent Manufacturing, 1998, 19: 209-224.
- [8] 贾丽会, 张修如. BP 算法分析与改进[J]. 计算机技术与发展, 2006, 16(10): 200-203.
- [9] 熊 凌, 赵明旺. 基于遗传算法的 BP 网络全局收敛的混合智能学习算法[J]. 武汉科技大学学报: 自然科学版, 2000, 23(2): 183-186.