

基于 Kernel-SOM 的非线性系统辨识及模型运行收敛性分析

於东军^① 郑宇杰^① 吴小俊^② 杨静宇^①

^①(南京理工大学计算机科学与技术学院 南京 210094)

^②(江南大学信息工程学院 无锡 214122)

摘要: 该文提出基于核 SOM(Kernel-SOM)的非监督非线性系统辨识方法。在辨识误差和系统初始输入误差同时存在的条件下,对 Kernel-SOM 辨识模型独立运行的收敛性进行了理论分析,并给出了辨识模型运行收敛的定理。数字仿真表明了所述方法的有效性及其收敛定理的正确性。

关键词: 核 SOM; 非线性系统; 辨识; 收敛性; 模式识别

中图分类号: TP271

文献标识码: A

文章编号: 1009-5896(2008)08-1928-04

Kernel-SOM Based Nonlinear System Identification and Model Running Convergence Analysis

Yu Dong-jun^① Zhen Yu-jie^① Wu Xiao-jun^② Yang Jing-yu^①

^①(School of Computer Science and Technology, Nanjing University of Science and Technology, Nanjing 210094, China)

^②(School of Information Engineering, Southern Yangtse University, Wuxi 214122, China)

Abstract: A Kernel-SOM based unsupervised nonlinear system identification algorithm is proposed. Analysis of the model running convergence of the proposed algorithm is performed, and the convergence theorem is proofed by considering both identification error and initial input error. Numerical simulation results demonstrate the effectiveness of the proposed identification algorithm and the correctness of the convergence theorem.

Key words: Kernel-SOM; Nonlinear system; Identification; Convergence; Pattern recognition

1 引言

学习算法的简单性、自适应、以及可视化等诸多良好的特性,使得自组织映射^[1](Self-Organizing Map, SOM)在众多领域得到广泛应用,例如:模式识别、非线性系统辨识、数据挖掘。SOM 是典型的自适应、非监督(无教师)学习算法,经过适当改造后可以完成有监督(教师)的学习。Barreto 在文献[2]中使用 SOM 实现了有监督的学习,并用于非线性系统的辨识。其核心思想是将每个输出节点的权值分为两部分:一部分对应于待学习模式的输入,用于寻找获胜节点;另一部分对应于待学习模式的输出(教师)。在学习阶段,输出节点的两部分权值均调整。在辨识阶段,获胜节点权值的输出部分用于估计待辨识系统的实际输出。文献[2]指出修改后的学习算法是收敛的。但是,该文并未讨论辨识模型独立运行时的收敛性问题。

在 SOM 中引入核函数,可以得到核 SOM(Kernel-SOM)。已有研究表明^[3,4],合适的 Kernel-SOM 更适合于解决非线性问题。在文献[2]的基础上,本文进一步拓展 SOM 至 Kernel-SOM 并用于非线性系统辨识,此外还研究辨识模

型独立运行的收敛性,给出了辨识模型运行收敛的条件。

2 Kernel-SOM

在经典的 SOM^[1]中,分为输入层和输出层。通常输出节点在输出层以二维的方式排列,输入节点和输出节点全互连。在 SOM 中引入核函数,可以得到基于核的 SOM(Kernel-SOM)。Kernel-SOM 首先将输入模式映射到一个高维的非线性空间,然后在映射后的高维空间中学习模式的分布规律。巧妙的是高维空间中的学习可以通过原始低维空间的学习间接完成得到。较之于经典的 SOM,合适的 Kernel-SOM 对于非线性强的模式分布可以有较好的学习结果^[3,4]。

设 $\Phi: \mathbf{x} \rightarrow \Phi(\mathbf{x}) \in F$ 为非线性映射,其中 $\mathbf{x} = (x_1, \dots, x_n)^T$ 为输入, F 为映射后的高维空间。定义高维空间中的距离度量如下:

$$KD(\mathbf{x}, \mathbf{y}) = \|\Phi(\mathbf{x}) - \Phi(\mathbf{y})\|^2 \quad (1)$$

满足 Mercer 条件^[5]的核函数可以定义如下:

$$K(\mathbf{x}, \mathbf{y}) = \Phi(\mathbf{x})^T \Phi(\mathbf{y}) \quad (2)$$

有

$$KD(\mathbf{x}, \mathbf{y}) = \|\Phi(\mathbf{x}) - \Phi(\mathbf{y})\|^2 = K(\mathbf{x}, \mathbf{x}) + K(\mathbf{y}, \mathbf{y}) - 2K(\mathbf{x}, \mathbf{y}) \quad (3)$$

用 $\mathbf{x}(z) = (x_1(z), \dots, x_n(z))$ 表示在第 z 步学习时 Kernel-SOM 的学习样本,输出层位于坐标 (k, j) 的输出节点的权值 $w_{kj}(z) = (w_{1kj}(z), w_{2kj}(z), \dots, w_{nkj}(z))$ 。 $N_{kj}(z)$ 为邻域函数,

2007-01-05 收到, 2007-09-24 改回

国家自然科学基金(60572034, 60632050),江苏省自然科学基金(BK2006081)和国家 863 计划项目(2006AA04Z237)资助课题

$\alpha(z)$ 为学习效率, m 为输出层的宽度。Kernel-SOM 学习算法可以简单描述如下:

(1) 使用小的随机数初始化所有的连接权。

(2) 从学习样本集中随机选择一个作为当前学习样本 $\mathbf{x}(z)$ 。

(3) 从输出节点中找出获胜节点, 亦即最接近当前学习样本的输出节点。

$$d_{k^*j^*} = \min_{1 \leq k, j \leq m} (d_{kj}) = \min_{1 \leq k, j \leq m} \|KD(\mathbf{x}(z), \mathbf{w}_{kj}(z))\| \quad (4)$$

(k^*, j^*) 为获胜节点在输出层的坐标, $k^*, j^* \in [1, m]$ 。

(4) 获胜节点及其邻域节点(由邻域函数确定)的权值进行更新

$$\mathbf{w}_{ij}(z+1) = \mathbf{w}_{ij}(z) + \alpha(z) \cdot N_{i^*j^*}(z, i, j) \cdot \frac{\partial KD(\mathbf{x}(z), \mathbf{w}_{ij}(z))}{\partial \mathbf{w}_{ij}(z)} \quad (5)$$

(5) 重复式(2)~式(4), 直到学习结果满足要求或是达到预先设定的学习次数。

3 基于 Kernel-SOM 的非线性系统辨识

给定非线性系统:

$$\mathbf{y}(t+1) = f(\mathbf{y}(t), \dots, \mathbf{y}(t-n_y+1), \mathbf{u}(t), \dots, \mathbf{u}(t-n_u+1)) \quad (6)$$

其中 $t \in [0, M]$, M 为有限正整数。 $\mathbf{u}(t) \in U \subset R^p$ 为确定的已知激励函数, $\mathbf{y}(t) \in Y \subset R^q$, n_u, n_y 分别是 \mathbf{y} 和 \mathbf{u} 最大延迟时间。

假设, 可以获取实际非线性系统的 N 个样本:

$$\underbrace{(\mathbf{y}(t_i), \dots, \mathbf{y}(t_i-n_y+1), \mathbf{u}(t_i), \dots, \mathbf{u}(t_i-n_u+1))}_{\text{input}}^T \rightarrow \underbrace{(\mathbf{y}(t_i+1))}_{\text{output}}^T \quad (7)$$

其中 $t_i \in [0, M]$, $1 \leq i \leq N$ 。

Kernel-SOM 在第 z 步学习时选中的样本用 $\mathbf{x}(z) = \begin{pmatrix} \mathbf{x}^{\text{input}}(z) \\ \mathbf{x}^{\text{output}}(z) \end{pmatrix}$ 表示:

$$\mathbf{x}^{\text{input}}(z) = (\mathbf{y}(t_i), \dots, \mathbf{y}(t_i-n_y+1), \mathbf{u}(t_i), \dots, \mathbf{u}(t_i-n_u+1))^T$$

$$\mathbf{x}^{\text{output}}(z) = (\mathbf{y}(t_i+1))^T$$

再令 $d^{\text{input}} = \text{Dim}(\mathbf{x}^{\text{input}}(z))$, $d^{\text{output}} = \text{Dim}(\mathbf{x}^{\text{output}}(z))$ 。

输出层位于坐标 (k, j) 的输出节点的权值 $\mathbf{w}_{kj}(z) = \begin{pmatrix} \mathbf{w}_{kj}^{\text{input}}(z) \\ \mathbf{w}_{kj}^{\text{output}}(z) \end{pmatrix}$, 其中

$$\mathbf{w}_{kj}^{\text{input}}(z) = (\mathbf{w}_{1kj}(z), \mathbf{w}_{2kj}(z), \dots, \mathbf{w}_{d^{\text{input}}kj}(z))^T$$

$$\mathbf{w}_{kj}^{\text{output}}(z) = (\mathbf{w}_{(d^{\text{input}}+1)kj}(z), \mathbf{w}_{(d^{\text{input}}+2)kj}(z), \dots, \mathbf{w}_{(d^{\text{input}}+d^{\text{output}})kj}(z))^T$$

那么使用 Kernel-SOM 来辨识非线性系统(式(6)), 获胜节点及其邻域节点(由邻域函数确定)的权值进行更新:

$$\mathbf{w}_{kj}^{\text{input}}(z+1) = \mathbf{w}_{kj}^{\text{input}}(z) + \alpha(z) N_{k^*j^*}(z, k, j) \cdot \frac{\partial KD(\mathbf{x}^{\text{input}}(z), \mathbf{w}_{kj}^{\text{input}}(z))}{\partial \mathbf{w}_{kj}^{\text{input}}(z)} \quad (8)$$

$$\mathbf{w}_{kj}^{\text{output}}(z+1) = \mathbf{w}_{kj}^{\text{output}}(z) + \alpha(z) N_{k^*j^*}(z, k, j) \cdot \frac{\partial KD(\mathbf{x}^{\text{output}}(z), \mathbf{w}_{kj}^{\text{output}}(z))}{\partial \mathbf{w}_{kj}^{\text{output}}(z)} \quad (9)$$

学习完毕后, 对于一个给定的输入 $\mathbf{x}^{\text{input}}$, Kernel-SOM 获胜节点的输出权值 $\mathbf{w}_{k^*j^*}^{\text{output}}$ 可以用来估计实际系统的输出, 即 $\hat{\mathbf{y}} = \hat{f}(\mathbf{x}^{\text{input}}, \Theta^*) = \mathbf{w}_{k^*j^*}^{\text{output}} \approx \mathbf{y} = f(\mathbf{x}^{\text{input}})$, 其中 $\hat{\mathbf{y}} = \hat{f}(\mathbf{x}^{\text{input}}, \Theta^*)$ 表示 Kernel-SOM 模型的输入输出映射关系, Θ^* 是 Kernel-SOM 模型参数的集合(权以及核函数参数)。

4 Kernel-SOM 模型的运行收敛性

对于式(6)所示的非线性系统, 令 $L_{\max} = \max \left\| \frac{\partial \mathbf{y}(t+1)}{\partial \mathbf{y}(t-n_y+i)} \right\|$, 其中 $1 \leq i \leq n_y, t \in [0, M]$, M 为有限正整数。

条件 1 L_{\max} 上有确界。

定理 1 非线性系统 $\mathbf{y}(t+1) = f(\mathbf{x}^{\text{input}}(t))$, f 满足条件 1。对应的 Kernel-SOM 模型为 $\hat{\mathbf{y}}(t+1) = \hat{f}(\hat{\mathbf{x}}^{\text{input}}(t), \Theta)$ 。如果实际系统 f 和 Kernel-SOM 模型 \hat{f} 的初始输入误差满足 $\|\mathbf{sy}(0) - \mathbf{s}\hat{\mathbf{y}}(0)\| < \delta$, 那么对于给定的 $\forall \varepsilon > 0$, 必定可以找到一个 Θ^* , 使得下式成立:

$$\max \|\mathbf{y}(t+1) - \hat{\mathbf{y}}(t+1)\| < \varepsilon \quad (10)$$

其中 $t \in [0, M]$, M 为有限正整数。

证明 使用文献[6]类似的方法, 用向量 $\mathbf{S}(t+1)$ 来表示实际系统运行到 $t+1$ 时刻的状态:

$$\mathbf{S}(t+1) = \begin{pmatrix} \mathbf{y}(t+1) \\ \mathbf{sy}(t) \end{pmatrix}, \text{ 其中 } \mathbf{sy}(t) = \begin{pmatrix} \mathbf{y}(t) \\ \dots \\ \mathbf{y}(t-n_y+1) \end{pmatrix}.$$

很显然, $\mathbf{sy}(t)$ 可以表示为

$$\mathbf{sy}(t) = \begin{pmatrix} 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & 0 & \vdots & 0 \\ \vdots & \dots & \ddots & \dots & \vdots \\ 0 & \dots & 0 & 1 & 0 \end{pmatrix}_{n_y \times (n_y+1)} \cdot \mathbf{S}(t) \triangleq g(\mathbf{S}(t)) \quad (11)$$

$$\text{再由 } \mathbf{su}(t) = \begin{pmatrix} \mathbf{u}(t) \\ \dots \\ \mathbf{u}(t-n_y+1) \end{pmatrix}, \text{ 易知 } \mathbf{y}(t+1) \text{ 是关于 } \mathbf{S}(t) \text{ 和 } \mathbf{su}(t) \text{ 的函数, 因此 } \mathbf{y}(t+1) \text{ 可以表示成:}$$

$$\mathbf{y}(t+1) \triangleq f(\mathbf{S}(t), \mathbf{su}(t)) \quad (12)$$

那么 $\mathbf{S}(t+1)$ 可以递归定义如下:

$$\mathbf{S}(t+1) = \begin{pmatrix} \mathbf{y}(t+1) \\ \mathbf{sy}(t) \end{pmatrix} = \begin{pmatrix} f(\mathbf{S}(t), \mathbf{su}(t)) \\ g(\mathbf{S}(t)) \end{pmatrix} \quad (13)$$

为此, 可以定义:

$$G(\mathbf{S}(t), \mathbf{su}(t)) \triangleq (\mathbf{S}(t+1))^T = \begin{pmatrix} f(\mathbf{S}(t), \mathbf{su}(t)) \\ g(\mathbf{S}(t)) \end{pmatrix}^T \quad (14)$$

类似地可以定义 $\hat{\mathbf{S}}(t)$ 来表示 Kernel-SOM 模型运行到 t

时刻的状态:

$$\hat{\mathbf{S}}(t+1) = \begin{pmatrix} \hat{\mathbf{y}}(t+1) \\ \mathbf{s}\hat{\mathbf{y}}(t) \end{pmatrix}, \quad \hat{G}(\hat{\mathbf{S}}(t), \mathbf{su}(t)) \triangleq (\hat{\mathbf{S}}(t+1))^T, \quad \text{其中}$$

$$\mathbf{s}\hat{\mathbf{y}}(t) = \begin{pmatrix} \hat{\mathbf{y}}(t) \\ \dots \\ \hat{\mathbf{y}}(t-n_y+1) \end{pmatrix}.$$

至此, 有

$$\begin{aligned} \|\mathbf{S}(t+1) - \hat{\mathbf{S}}(t+1)\| &= \|(\mathbf{S}(t+1))^T - (\hat{\mathbf{S}}(t+1))^T\| \\ &= \|G(\mathbf{S}(t), \mathbf{su}(t)) - \hat{G}(\hat{\mathbf{S}}(t), \mathbf{su}(t))\| \\ &\leq \underbrace{\|G(\mathbf{S}(t), \mathbf{su}(t)) - G(\hat{\mathbf{S}}(t), \mathbf{su}(t))\|}_{(I)} \\ &\quad + \underbrace{\|G(\hat{\mathbf{S}}(t), \mathbf{su}(t)) - \hat{G}(\hat{\mathbf{S}}(t), \mathbf{su}(t))\|}_{(II)} \end{aligned}$$

由中值定理, 必定存在:

$$\bar{\mathbf{S}}(t) = \lambda \mathbf{S}(t) + (1-\lambda) \cdot \hat{\mathbf{S}}(t), \quad 0 < \lambda < 1 \quad (15)$$

使得

$$(I) = \left\| \frac{\partial G(\mathbf{S}(t), \mathbf{su}(t))}{\partial \mathbf{S}(t)} \Big|_{\bar{\mathbf{S}}(t)} (\mathbf{S}(t) - \hat{\mathbf{S}}(t)) \right\| \quad (16)$$

结合条件 1, 容易证明:

$$\begin{aligned} &\left\| \frac{\partial G(\mathbf{S}(t), \mathbf{su}(t))}{\partial \mathbf{S}(t)} \Big|_{\bar{\mathbf{S}}(t)} \right\|_F \\ &\leq \sqrt{n_y \cdot 1^2 + \left(\frac{n_y \cdot (n_y + 1)}{2} + n_y \right) L_{\max}^2} \triangleq L_m \quad (17) \end{aligned}$$

$$\text{令 } \bar{G}(\hat{\mathbf{S}}(t), \mathbf{su}(t)) = G(\hat{\mathbf{S}}(t), \mathbf{su}(t)) - \hat{G}(\hat{\mathbf{S}}(t), \mathbf{su}(t))$$

那么, 有

$$\begin{aligned} \bar{G}(\hat{\mathbf{S}}(t), \mathbf{su}(t)) &= G(\hat{\mathbf{S}}(t), \mathbf{su}(t)) - \hat{G}(\hat{\mathbf{S}}(t), \mathbf{su}(t)) \\ &= (0, 0, \dots, f(\hat{\mathbf{S}}(t), \mathbf{su}(t)) - \hat{f}(\hat{\mathbf{S}}(t), \mathbf{su}(t), \Theta)) \quad (18) \end{aligned}$$

由万能逼近定理, 对于 $\forall \varepsilon_e > 0$, 总能找到一个 Θ^* , 使得

$$(II) = \|\bar{G}(\hat{\mathbf{S}}(t), \mathbf{su}(t))\| < \varepsilon_e \quad (19)$$

至此, 有

$$\|\mathbf{S}(t+1) - \hat{\mathbf{S}}(t+1)\| < (L_m)^{t+1} \|\mathbf{S}(0) - \hat{\mathbf{S}}(0)\| + \varepsilon_e \left(\sum_{i=0}^t (L_m)^i \right)$$

再令

$$\delta \triangleq \left(\varepsilon - \varepsilon_e \left(\sum_{i=0}^t (L_m)^i \right) \right) \frac{1}{(L_m)^{t+1}} \quad (20)$$

所以, 对于 $\forall \varepsilon > 0$, 只要设定:

$$\|\mathbf{S}(0) - \hat{\mathbf{S}}(0)\| < \delta \quad (\text{而 } \|\mathbf{sy}(0) - \mathbf{s}\hat{\mathbf{y}}(0)\| \leq \|\mathbf{S}(0) - \hat{\mathbf{S}}(0)\|), \quad \text{所以}$$

$$\|\mathbf{sy}(0) - \mathbf{s}\hat{\mathbf{y}}(0)\| < \delta, \quad \text{就必定有 } \|\mathbf{S}(t+1) - \hat{\mathbf{S}}(t+1)\| < \varepsilon.$$

又显然有

$$\|\mathbf{y}(t+1) - \hat{\mathbf{y}}(t+1)\| \leq \|\mathbf{S}(t+1) - \hat{\mathbf{S}}(t+1)\| \quad (21)$$

即必定有

$$\|\mathbf{y}(t+1) - \hat{\mathbf{y}}(t+1)\| < \varepsilon \quad (22)$$

再由 t 得任意性, 立即得

$$\max \|\mathbf{y}(t+1) - \hat{\mathbf{y}}(t+1)\| < \varepsilon \quad (23)$$

证毕

5 数字仿真

本小节使用文献[7]中给出的一个典型非线性系统来验证 Kernel-SOM 辨识模型的有效性及其收敛定理的正确性。

$$y(t+1) = \frac{y(t)y(t-1)(y(t)+2.5)}{1+y^2(t)+y^2(t-1)} + u(t) \quad (24)$$

$u(t) = \sin(2\pi t/25)$ 为确定的激励函数。系统初始状态值 $y(-1) = 0.5, y(0) = 0.9$ 。

数据集 从式(24)所示的非线性系统采集 100 个样本, 前 50 个样本用于学习, 后 50 个样本用作测试。

实验结果 (1)分别构建输出层规模为 $5 \times 5, 10 \times 10$ 和 15×15 的 SOM 及 Kernel-SOM, 在数据集集中的训练和测试结果见表 1。

表 1 SOM 与 Kernel-SOM 性能比较

SOM 规模		均方差		
		5×5	10×10	15×15
SOM	训练样本	0.012821	0.000368	0.000152
	测试样本	0.016030	0.000587	0.000192
Kernel-SOM	训练样本	0.010140	0.000143	0.000098
	测试样本	0.011712	0.000432	0.000102

注: 表中给出的数据均为 10 次运行的平均值。最大学习次数 $z_{\max} = 10000$; 使用正方形邻域函数, 在第 z 次学习时, 正方形的宽度为 $\rho(z)$, 并按照下式更新:

$$\rho(z+1) = \left\lfloor \rho_0 (\rho_f / \rho_0)^{z/z_{\max}} \right\rfloor, \quad \text{其中 } \rho_f = 0.4, \quad \rho_0 = \lfloor D/2 \rfloor.$$

D 为输出层的宽度; 第 z 次学习时的学习效率, 按照下式更新: $\alpha(z+1) = \alpha_0 (\alpha_f / \alpha_0)^{z/z_{\max}}$, 其中 $\alpha_f = 0.04$, $\alpha_0 = 0.9$ 高斯核函数宽度, $\sigma = \sqrt{2}$ 。

(2)图 1 和图 2 分别给出了输出层规模为 5×5 和 10×10 的 Kernel-SOM 独立运行的结果。图中*所示的曲线为实际系统的运行曲线; □为 Kernel-SOM 模型独立运行的运行曲线。为了验证定理 1 的有效性, 特别设置 Kernel-SOM 模型的初始输入值为 $\hat{\mathbf{y}}(-1) = 0.8, \hat{\mathbf{y}}(0) = 1.2$, 不同于实际系统 $y(-1) = 0.5, y(0) = 0.9$ 。

分析 由表 1 可以发现: 在不同的规模下, Kernel-SOM 的性能总是优于 SOM。此外, 当网络规模不够大时(例如 5×5), 即使学习次数足够(10000), 学习精度仍不够理想。原因是当网络规模小时, 其中的输出权个数较少, 因此对输出论域的划分粒度粗糙, 导致精度不高; 随着网络规模的增大, 对输出论域的划分粒度逐步精细, 精度也随之上升。图 1 和图 2 验证了定理 1 的正确性。Kernel-SOM 和实际系统存在逼近误差, 即使初始输入不同, 在一定的条件下, 仍能保证

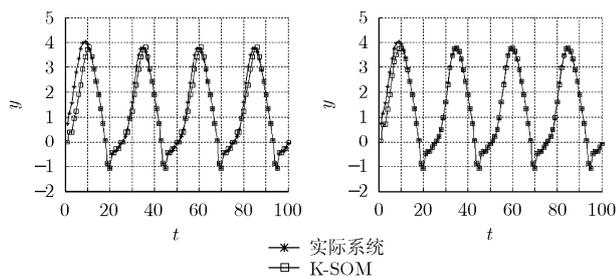


图 1 5×5 Kernel-SOM
独立运行结果图

图 2 10×10 Kernel-SOM
独立运行结果图

Kernel-SOM 独立运行收敛。图 1 中, 逼近误差较大, 但是在初始输入不同的条件下, Kernel-SOM 仍良好地跟踪了实际系统。在图 2 中, 由于逼近精度较高, 在初始输入不同时, Kernel-SOM 除了一开始有所震荡, 几乎复现了实际系统。

6 结论

使用非监督学习进行非线性系统辨识是神经网络领域研究的一个热点问题。本文给出了运用 Kernel-SOM 模型辨识非线性系统的方法。考虑到实际应用中固有的辨识误差和初始输入误差的并存, 文中讨论了 Kernel-SOM 辨识模型的运行收敛性问题, 并以定理的形式给出运行收敛的条件。典型非线性系统的数字仿真结论验证了本文所述方法的有效性。

参考文献

[1] Kohonen T. Self-organization map. *Proc. IEEE*, 1990, 78(9):

1464-1480.

- [2] Barreto G A and Aluizio A F R. Identification and control of dynamical systems using the self-organizing map. *IEEE Trans. on Neural Networks*, 2004, 15(5): 1244-1259.
- [3] Yu Dong-jun, *et al.* Kernel-SOM based visualization of financial time series forecasting. International Conference on innovative computing, information and control. Beijing, 2006, Volume II: 470-473.
- [4] Pan Zhisong, Chen Songcan, and Zhang Daoqiang. A Kernel-based SOM classification in input space. *Acta Electronica Sinica*, 2004, 32(2): 227-231.
- [5] Scholkopf B, Burges C J C, and Smola A J. *Advances in Kernel Methods - Support Vector Learning* [M]. Cambridge, MA, The MIT Press, 1999: 255-268.
- [6] Pao Xiaohong, *et al.* Model error analysis in nonlinear system identification using neural networks (I). *Control and Decision*, 1997, 12(5): 20-25.
- [7] Lin C T. *Neural Fuzzy Systems*. New York: Prentice-Hall Press. 1997, Chapter 3.

於东军: 男, 1975 年生, 副教授, 硕士生导师, 研究方向为神经网络、模糊系统、模式识别。

郑宇杰: 男, 1978 年生, 博士, 研究方向为人脸识别、模式识别。

吴小俊: 男, 1967 年生, 教授, 博士生导师, 研究方向为神经网络、模式识别。

杨静宇: 男, 1941 年生, 教授, 博士生导师, 研究方向为机器视觉、模式识别。