

方向基概率神经网络的模式识别*

罗雄彪 陈铁群 万 英
(华南理工大学机械工程学院 广州 510640)

摘要:针对方向基函数神经网络与概率神经网络的优缺点,提出了方向基概率神经网络的基本模型,并讨论了它用作模式分类时的相关原理及算法。以焊缝中裂纹和气孔缺陷的分类为例,用方向基概率神经网络对缺陷进行模式识别。结果表明:该网络应用于模式识别,较方向基函数神经网络和概率神经网络不仅在速度上有较大的提高,而且在分类性能上也有明显的改善。

关键词:方向基函数神经网络 概率神经网络 融合 模式识别 方向基概率神经网络

中图分类号: TP183

0 前言

方向基函数神经网络(DBFNN)模型是一种两层神经网络,输出层神经元的权值矢量可视为中心矢量,而输入的样本可视为输入样本矢量。如果输出层神经元的非线性函数和输入层至输出层的权值矢量满足一定的条件,神经元的输出则可以实现输入样本矢量与权值矢量夹角的函数。自从 S. J. Wang 等^[1]提出 DBFNN 以来, DBFNN 已经广泛地应用于模式识别、控制系统以及图像处理等领域;同时已经研究出 CASSANDRA-1 预言神一号小型神经计算机,在硬件上实现了方向基函数(DBF)模式分类器^[2]。

概率神经网络(PNN)是从模式的概率分布出发,按照最大后验概率分类规则,对网络输出的最大似然值来判决模式识别的,使得分类所造成的平均损失最小^[3]。它是由 D. F. Specht^[4]于 1989 年提出的,利用神经网络大规模并行处理的特点,给出了若干 Parzen 窗函数估计的 PNN 结构,该网络对应的权值是模式样本的分布,网络不需要训练,因而能够满足训练上的实时处理要求。这种网络已较广泛地应用于非线性滤波、模式识别、联想记忆与概率密度估计等领域当中,但是它的权值学习算法及有关的理论有待于进一步的研究^[5]。

通过对 DBFNN 与 PNN 两种模型比较,对它们的优点进行融合,探讨方向基概率神经网络(DBPNN)模型结构、分类原理和算法。

1 DBFNN 与 PNN 比较

DBFNN 与 PNN 都是前馈神经网络(FNN),其

隐层神经元的传输函数可以选择满足 Parzen 窗函数的所有势函数。假设 DBFNN 的所有隐层神经元的中心矢量是 V_j , $j=1, 2, 3, \dots, H$, 基函数角度宽度是 r_j , 方向基函数为

$$h_j(X) = \exp\{-[1 - X^T V_j / (|X| |V_j|)] / r_j^2\} \quad (1)$$

式中 X ——样本矢量(x_1, x_2, \dots, x_n)^T

H ——隐层神经元的总数

把 $h_j(X)$ 函数作为 Parzen 窗函数,则 DBFNN 和 PNN 的第 i 个节点的输出 O_i^d 、 O_i^p 可以分别表示为

$$\begin{aligned} O_i^d &= \sum_{k=1}^H w_{ik} h_j(X) = \sum_{x_k \in w_i} w_{ik} h_j(X) + \sum_{x_k \notin w_i} w_{ik} h_j(X) \\ O_i^p &= \sum_{k=1}^{H_i} h_j(X) \approx p(X/w_i) \end{aligned} \quad (2)$$

式中 H_i ——PNN 中第 i 类别对应的隐层神经元总数, $i=1, 2, \dots, c$, c 为模式类别总数

w_{ik} ——第 i 个神经元节点对应的第 k 个连接权矢量

w_i ——第 i 个模式类别集合矢量

$p(X/w_i)$ ——类条件概率密度

从式(2)可以看出 DBFNN 和 PNN 有以下差异。

(1) PNN 的隐层神经元数正好等于参加训练的总样本数,而 DBFNN 却小于或等于参加训练的总样本数。

(2) DBFNN 的隐层节点与输出节点之间的连接权矢量需要反复训练才能形成各个类别模式间的判别表面,而 PNN 的隐层节点与输出节点之间的连接权矢量无需训练(取常数值 1),模式类别间的判别表面由类别模式的类条件概率密度决定。

(3) DBFNN 需外监督信号来监督训练,而 PNN 靠类别属性标记进行自监督分类。

(4) PNN 直接测试,无需训练,而 DBFNN 需要反复训练。

* 广东省科技计划基金资助项目(2004A11303001)。20050311 收到初稿, 20050818 收到修改稿

从式(2)也可以明显看出: O_i^d 不但与第 i 个类别样本特性有关系, 而且还与其他类别的样本特性有关, 即 DBFNN 考虑了不同类别模式间的重叠和交错影响; O_i^p 只与第 i 类别的样本特性有关, 没有考虑不同类别模式间的重叠和交错影响, 所以虽然无需训练, 但其性能是有限的。正好相反, DBFNN 的整体性能很好, 训练时间较长(相对 PNN 网络来说)。所以提出一种能利用两种模型优点进行融合的 DBPNN 模型。

2 DBPNN 模型描述

2.1 模型结构

DBPNN 模型主要包括 3 层, 即输入层、隐层和输出层, 图 1 给出这种模型结构的示意图。

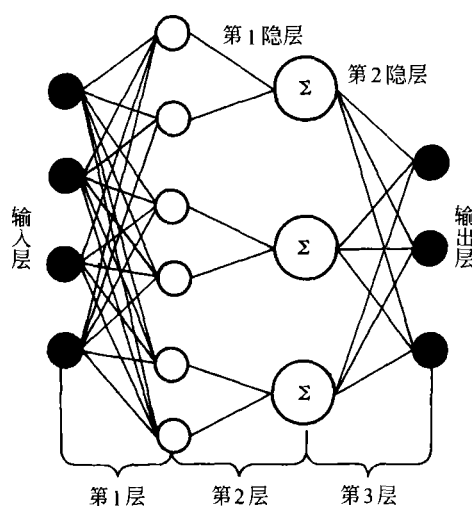


图 1 DBPNN 结构示意图

DBPNN 模型的第 1 层等同于 DBFNN 的第 2 层, 其第 1 隐层的节点传输函数与 DBFNN、PNN 一样, 都是满足 Parzen 窗函数的方向基函数。该方向基函数的中心矢量 V_j , 即代表第 1 层连接权矢量

$$w_i^{(1)} = V_j \quad w_i^{(1)} \in \mathbf{R}^n \quad i=1, 2, \dots, H_1$$

式中 H_1 ——表示第 1 隐层的节点数

第 1 层连接权矢量 $w_i^{(1)}$ 可以事先确定或通过自适应学习来选择。

第 2 层等同于 PNN 的第 2 层, 其第 2 隐层为求和与运算节点, 该层节点与第 1 层节点, 按类别进行有选择的连接, 且其中的连接权矢量都取 1 的值

$$w_i^{(2)} = 1 \quad i=1, 2, \dots, H_2$$

$w_i^{(2)}$ 矢量维数的大小, 要视与第 1 隐层的节点数 H_1 的多少来定。

第 3 层等同于 DBFNN 的第 2 层, 其输出节点与 DBFNN 一样, 都是线性的。如果单从这一层来

看, 相当于单层线性感知器。该层权矢量需要根据训练集样本的特性自适应学习得到, 通常的学习算法包括广义 δ -规则反向传播, 递推最小二乘法反向传播等。

2.2 隐层神经元总数的选择

从神经网络模式分类器的机理来看, 网络的隐层一般是起模式变换与特征提取的作用, 所以神经元节点数的选择应该根据被分类模式的特性来进行。对于空间上连通的或者稠密的模式, 即各种类别之间几乎没有产生重叠与交错影响, 对于每一个类别, 可以选择对应连通区域中的所有样本对应的典型样本, 作为它在第 1 隐层的中心矢量。假设 c 个类别分别对应的典型样本数为 $M^{(1)}, M^{(2)}, \dots, M^{(c)}$, 则第 1 隐层与第 2 隐层神经元数 H_1, H_2 可分别表示为

$$H_1 = \sum_{k=1}^c M^{(k)} \quad H_2 = c$$

对于空间上不连通的或者较稀疏的模式(一般是这种情况), 对于每个类别可以将模式样本按不同的连通区域或分区提取典型样本。作为该类别在第 1 隐层中的中心矢量, 假设 c 个类别分别对应的典型样本数为

$$\begin{aligned} w_1: & M_1^{(1)}, M_2^{(1)}, \dots, M_{n_1}^{(1)} \\ w_2: & M_1^{(2)}, M_2^{(2)}, \dots, M_{n_2}^{(2)} \\ & \vdots \\ w_k: & M_1^{(k)}, M_2^{(k)}, \dots, M_{n_k}^{(k)} \\ & \vdots \\ w_c: & M_1^{(c)}, M_2^{(c)}, \dots, M_{n_c}^{(c)} \end{aligned}$$

其中 n_k 为第 k 个类别对应的连通区域, 则第 1 与第 2 隐层神经元数 H_1, H_2 分别为

$$H_1 = \sum_{k=1}^c \sum_{j=1}^{n_k} M_j^{(k)} \quad H_2 = \sum_{k=1}^c n_k$$

记 $L_i = \sum_{k=1}^c n_k$, 且假设 $L_0=0$ 。

3 DBPNN 分类原理与算法研究

3.1 分类原理

设一个神经元的中心矢量为 V_j , 样本矢量为 X 。如果神经元的激活函数为硬限幅形式

$$h_j(X) = H[\cos(X, \wedge V_j) - \theta_j] = H(XV_j^T - \theta)$$

$$H(X) = \begin{cases} 1 & X \geq 0 \\ 0 & X < 0 \end{cases}$$

式中 $(X, \wedge V_j)$ —— X 与 V_j 之间的夹角

此时第 1 隐层的第 i 个神经元输出为 O_i^1

$$O_i^1 = H(XV_j^T - \theta) \quad (3)$$

由点积公式

$$\frac{XV_j^T}{|X||V_j^T|} = \cos \alpha$$

α 为多维欧氏空间矢量 X 与 V_j^T 的夹角, 当 X 与 V_j^T 为量纲一矢量, 即量纲一化到超球面时, $XV_j^T = k \cos \alpha$, k 为某一常数, 则式(3)进一步表示为^[6]

$$\text{当 } k \cos \alpha \geq 0 \text{ 时 } O_i^1 = 1$$

$$\text{当 } k \cos \alpha < 0 \text{ 时 } O_i^1 = 0$$

对于 c 个类别的模式识别问题, 假设模式分布为最一般形式的不连通情况, 第 2 隐层的神经元输出端第 k 个节点输出 O_k^2 为^[7]

$$O_k^2 = \frac{1}{\varphi(k)} \sum_{j=1}^{\varphi(k)} h_j(X) \quad (4)$$

$$\varphi(k) = M_{k-L_{i-1}}^{(i)}$$

$$L_{i-1} < k \leq L_i \quad i=1, 2, \dots, c$$

式中 $\varphi(k)$ ——第 $k-L_{i-1}$ 连通区域中含有第 i 类典型样本数

假设在第 2 层中的第 i 个节点对应的类别标记为 t , 则 O_i^2 相当于第 i 个节点对应的第 t 个类别的条件概率密度, 即式(4)可进一步表示为

$$O_i^2 \approx p_i(X/w_i^{(2)})$$

设来自 c 个类别样本的训练样本数分别为 N_1, N_2, \dots, N_c , 记 $\sum_{i=1}^c N_i = N$, 则 DBPNN 输出层的输出端第 m 个节点的输出 O_m^3 为

$$O_m^3 = \sum_{k=1}^{H_2} w_{mk}^{(3)} O_k^2 = \sum_{k=1}^{H_2} \frac{w_{mk}^{(3)}}{\varphi(k)} \sum_{j=1}^{\varphi(k)} h_j(X) \quad (5)$$

若 DBPNN 输出端对应的外监督信号为 0、1 二进制形式, 根据式(5), 可以写出 N 个训练样本在输出端第 3 层网络所遵从的方程为

$$\begin{bmatrix} O_{11}^{(1)} & O_{12}^{(1)} & \cdots & O_{1H_2}^{(1)} \\ \vdots & \vdots & & \vdots \\ O_{N_1 1}^{(1)} & O_{N_1 2}^{(1)} & \cdots & O_{N_1 H_2}^{(1)} \\ \vdots & \vdots & & \vdots \\ O_{11}^{(c)} & O_{12}^{(c)} & \cdots & O_{1H_2}^{(c)} \\ \vdots & \vdots & & \vdots \\ O_{N_c 1}^{(c)} & O_{N_c 2}^{(c)} & \cdots & O_{N_c H_2}^{(c)} \end{bmatrix} \times$$

X

$$\underbrace{\begin{bmatrix} w_{11}^{(3)} & w_{12}^{(3)} & \cdots & w_{c1}^{(3)} \\ w_{12}^{(3)} & w_{22}^{(3)} & \cdots & w_{c2}^{(3)} \\ \vdots & \vdots & & \vdots \\ w_{1H_2}^{(3)} & w_{2H_2}^{(3)} & \cdots & w_{cH_2}^{(3)} \end{bmatrix}}_{W^{(3)}} = \underbrace{\begin{bmatrix} 1 & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 1 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}}_O$$

此式可以进一步表示为

$$O = XW^{(3)} \quad (6)$$

定义误差代价函数

$$E(W^{(3)}) = \|O - XW^{(3)}\|$$

该误差代价函数在权值空间 $W^{(3)}$ 上是凸函数, 其 Hessian 矩阵 $2X^T X \geq 0$ 为半正定矩阵^[8], 当 $H_2 \leq N$ 时, $X^T X > 0$, 即变为正定矩阵, 令 $\partial E(W^{(3)}) / \partial W^{(3)} = 0$, 因此 $E(W^{(3)})$ 一定将获得全局最小值解, 则 $W^{(3)} = (XX^T)^{-1} X^T O$; 若 $R(O) \subseteq R(X)$, 将获得严全局最小值解。所以, 使用有关算法可得到 DBPNN 的第 3 层连接权矩阵 $W^{(3)}$ 。

3.2 算法研究

DBPNN 模型的第 1 层和第 2 层连接权值分别为 V_j 和 1, 第 3 层相当于单层感知器网络, 具有线性处理单元的网络, 对于其连接权矩阵 $W^{(3)}$, 提出利用递推最小二乘法反向传播算法(RLS-BP)进行训练得到。

DBPNN 第 3 层的输入与输出神经元节点数分别为 H_2 和 c 。先定义至第 z 个迭代时刻的加权误差目标函数为^[7]

$$\begin{cases} J(z) = \frac{1}{2} \sum_{t=1}^z \lambda^{z-t} \sum_{j=1}^c \varepsilon_j^2(t) \\ \varepsilon_j(t) = D_j(t) - O_j(t) \end{cases} \quad (7)$$

式中 λ ——加权遗忘因子

t ——输入模式矢量的次数标记

$\varepsilon_j(t)$ ——输出层第 j 个节点在第 t 个模式输入时对应的误差信号

$D_j(t)$ ——输出层第 j 个节点在第 t 个模式输入时对应的期望输出

$O_j(t)$ ——输出层第 j 个节点在第 t 个模式输入时对应的实际输出

其中 λ 的作用是将过去样本对当前估值的影响逐渐忘掉, 使所估计的参数尽量反映当前时刻样本

特性, 通常 λ 取值范围为 $0 < \lambda < 1$ 。

根据式(6), DBPNN 输出端实际输出 $O_j(t)$ 可以表示为

$$O_j(t) = X(t)W_j^{(3)}(z)$$

式中

$$W_j^{(3)}(z) = [w_{j1}^{(3)} \quad w_{j2}^{(3)} \quad \cdots \quad w_{jH_2}^{(3)}]$$

$$X(t) = [x_1(t) \quad x_2(t) \quad \cdots \quad x_{H_2}(t)]$$

将 $J(z)$ 对 $W_j^{(3)}(z)$ 微分, 并且令 $W_j^{(3)}(z) = \hat{W}_j^{(3)}(z)$, 得

$$\frac{\partial J(z)}{\partial \hat{W}_j^{(3)}} = -\sum_{t=1}^z \lambda^{z-t} \frac{\partial O_j(t)}{\partial \hat{W}_j^{(3)}} \varepsilon_j(t)$$

令此式等于 0 可得

$$\sum_{t=1}^z \lambda^{z-t} X(t)X^T(t)\hat{W}_j^{(3)}(z) = \sum_{t=1}^z \lambda^{z-t} D_j(t) \quad (8)$$

将(8)式写成

$$\hat{U}(z)\hat{W}_j^{(3)}(z) = \hat{G}(z)$$

$$\hat{U}(z) = \sum_{t=1}^z \lambda^{z-t} X(t)X^T(t) = \lambda \hat{U}(z-1) + X(z)X^T(z)$$

$$\hat{G}(z) = \sum_{t=1}^z \lambda^{z-t} X(t)D_j(t) = \lambda \hat{G}(z-1) + X(z)D_j(z)$$

记 $P(z) = \hat{U}^{-1}(z)$, 称逆相关矩阵。利用矩阵求逆引理可得 DBPNN 第 3 层连接矢量的递推最小二乘关系式如下

$$g(z) = \frac{P(z-1)X(z)}{\lambda + X^T(z)P(z-1)X(z)} \quad (9)$$

$$P(z) = \frac{1}{\lambda} [P(z-1) - g(z)X^T(z)P(z-1)] \quad (10)$$

$$\hat{W}_j^{(3)}(z) = \hat{W}_j^{(3)}(z-1) + g(z)[D_j(z) - X^T(z)\hat{W}_j^{(3)}(z-1)] \quad (11)$$

$$J(z) = \lambda J(z-1) + \frac{1}{2} \sum_{j=1}^{H_2} [D_j(z) - X^T(z)\hat{W}_j^{(3)}(z-1)]^2 \quad (12)$$

式中 $g(z)$ —— z 时刻的卡尔曼增益矢量

式(9)~(12)就是 DBPNN 模型在已知输入模式、期望输出时所得到的第 3 层连接矢量递推最小二乘迭代公式。连接权矢量 $W^{(3)}$ 算法步骤如图 2 所示。

在讨论上述训练算法时, 都是假设隐单元中心矢量 V_j 是固定不变的, 而且还假定激活函数窗宽度(形状参数) r_j 也是取常数。实际上, r_j 表示感受的“视野”宽度大小, 从拓扑上来说, 它影响感受空间的拓扑分布, 所以 r_j 的取值大小将影响分类器的性能, 甚至影响网络的收敛速度。下面讨论激活函数窗宽度(形状参数) r_j 的梯度下降自适应迭代选择算法^[9]。

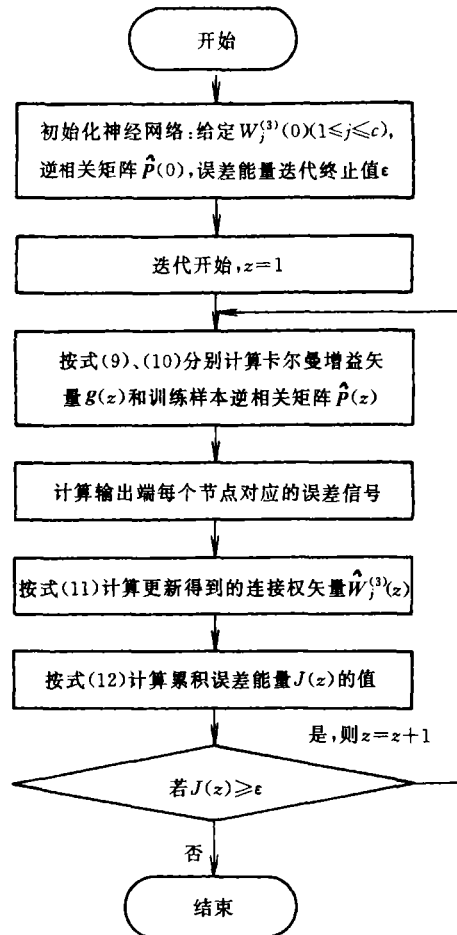


图 2 训练算法

假设式(1)为激活函数, 用来作为 DBPNN 隐单元的传输函数; 在 DBPNN 的输出端同样定义像式(7)那样的加权输出误差目标函数, 结合式(5), 由 $J(z)$ 对 $r_j(z)$ 求负梯度可得

$$\delta_j(z) = -\frac{\partial J(z)}{\partial r_j(z)} = \lambda \delta_j(z-1) + \left\{ \sum_{z=1}^{H_2} [D_z(t) - O_z(t)] \times \right.$$

$$\left. \sum_{k=1}^{H_2} \frac{w_{jk}^{(3)}}{\varphi(k)} \sum_{j=1}^{\varphi(k)} \exp\{-[1 - \cos(X, \hat{V}_j)]/r_j^2\} \right\} \times$$

$$\frac{2[1 - \cos(X, \hat{V}_j)]}{r_j^3(z)}$$

由此负梯度, 可得到 $r_j(z)$ 的梯度迭代公式

$$r_j(z) = r_j(z-1) + \eta r_j(z) + \zeta [r_j(z-1) - r_j(z-2)]$$

式中 η —— 学习步长 ($0 < \eta < 0.1$)

ζ —— 动量因子 ($0 < \zeta < 0.001$)

这就是 $r_j(z)$ 的梯度下降自适应迭代算法。

此外, 隐层神经元中心矢量 V_j 也可以进行学习算法修正, 通常的方法有 Kohonen 自组织特征映射方法、K-均值聚类法和自适应共振理论方法等, 这里不再讨论。

4 试验仿真结果分析

为了验证 DBPNN 应用于模式识别的有效性,对焊缝中裂纹和气孔两类缺陷进行了分类研究。

假设 DBPNN 的输入节点(训练样本维数)、隐层神经元数和输出节点分别为 N 、 M 、 L , 网络所得到的总训练样本数为 S ; 所有训练样本都被使用作为 DBPNN 隐神经元的中心矢量, DBPNN 模型中的第 2 隐节点数与待分类的模式类别数相等, 而且总样本中有一半被用于 DBPNN 的隐神经元中心。对于裂纹和气孔两类样本, 每类模式分别提取 30 个训练样本, 在这 30 个样本中再抽取 20 个作为隐神经元中心矢量, 则共需 40 个隐神经元, 即 $M=40$; 显然, $N=30$, $L=2$, $S=60$, 那么可以得到 DBFNN、PNN 和 DBPNN 三种网络单次迭代和测试的计算复杂度(乘除运算)和训练速度(每单位间隔乘除运算次数), 如表 1 所示。根据上述缺陷样本, 使用 RLS-BP 算法来训练 DBPNN, 最后使用收敛权值来测试剩余模板样本。

表 1 计算复杂度与训练速度比较

网络	计算复杂度		训练速度	
	训练	测试	训练	测试
DBFNN	$S^2+2S+SN+L+7$	$SN/2+S$	5 529	960
PNN	无	SN	无	1 800
DBPNN	$L^2+SN/4+L+7$	$SN/4+L$	463	452

图 3 给出了 DBFNN 与 DBPNN 在终值为 10^{-3} 时的误差收敛曲线的比较, 不过两者的横轴(迭代时间轴)单位时不同的, DBFNN 对应的单位为每单位间隔需要 5 529 次乘除运算, 而 DBPNN 为 463 次, 所以从训练速度上来说, DBPNN 比 DBFNN 要快得多。

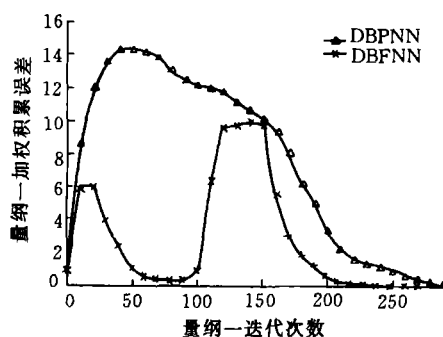


图 3 DBFNN 与 DBPNN 误差收敛曲线比较

图 4 分别给出了 PNN、DBFNN 和 DBPNN 三种网络对两种缺陷的测试判决曲线比较, 其中 PNN 使用的形状参数 $r=0.07$, DBFNN 的为 $r=0.6$, 而 DBPNN 的为 $r=0.4$ 。该曲线表示每类缺陷在网络输

出端相应的输出节点的测试结果。从图 4 可以看出, DBPNN 模型的置信度高于 PNN 网络的置信度, 低于而 DBFNN 的置信度。

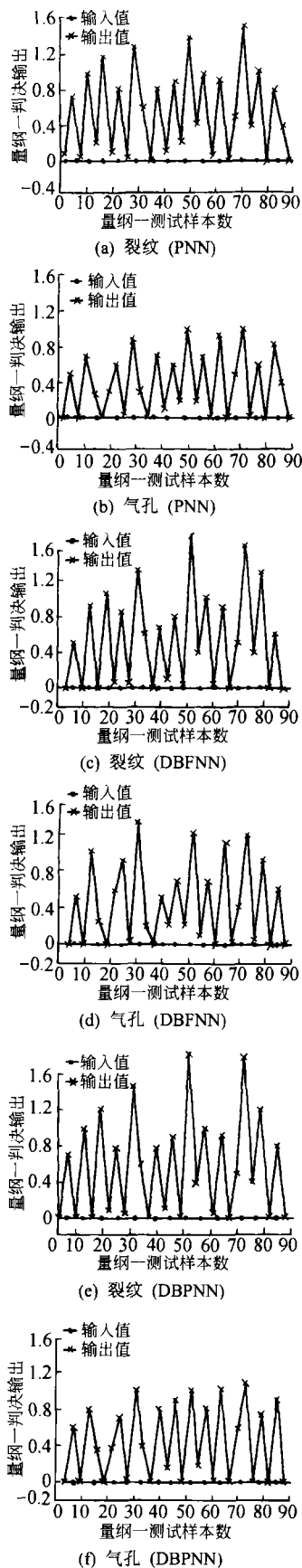


图 4 PNN、DBFNN 和 DBPNN 缺陷判决置信度曲线

根据 DBPNN 的最大后验概率准则, 所得到的识别率如表 2 所示。从表中仿真结果可以看到, DBPNN 在识别率上低于 DBFNN, 高于 PNN, 而在训练速度方面明显优于后两种网络。特别地, DBPNN 网络结构降低了网络的规模, 可为以后使用硬件加以实现提供方便。

表 2 缺陷识别结果

网络	裂纹			气孔			平均 识别率
	样本 个数	拒识 个数	识别率 $r/\%$	样本 个数	拒识 个数	识别 率 $r/\%$	
DBFNN	30	2	93.3	30	1	96.7	95.0
PNN	30	4	86.7	30	3	90.0	88.3
DBPNN	30	3	90.0	30	2	93.3	91.7

5 结论

通过对 DBFNN 与 PNN 各自优点的融合, 描述了 DBPNN 的基本结构模型, 并讨论了它用作模式分类时的相关原理及算法。最后对焊缝中裂纹和气孔缺陷进行了模式分类, 试验表明该模型的建立不仅在训练速度上有较大的提高, 而且在分类性能上有了明显的改善, 为缺陷模式分类问题的解决提供了一个有效的手段。

参 考 文 献

- 1 Wang S J, Shi J P. Direction-basis-function neural networks. In: Proceedings of IJCN'99, Washington, DC, 1999
- 2 Wang S J, et al. On the hardware for artificial neural networks and neurocomputer. Journal of Shenzhen University Science & Engineering, 1997, 14(1): 8~13
- 3 边肇祺, 张学工. 模式识别(第二版). 北京: 清华大学出版社, 2000
- 4 Specht D F. PNN and polynomial adlineas complementary techniques for classification. Transaction on Neural Network, 1990, 1(1): 111~121
- 5 罗发龙, 李衍达. 神经网络信号处理. 北京: 电子工业出版社, 1993
- 6 Wang X G, Cao W M. Property and implementation based on directional basis function neural network. In: Chinese Association of Artificial Intelligence, The Development of

Chinese Artificial Intelligence. The Ninth National Conference on Artificial Intelligence, Beijing, 2001, Beijing: Beijing University of Posts and Telecommunications Press, 2001: 450~454

- 7 黄德双. 神经网络模式识别系统理论. 北京: 电子工业出版社, 1996
- 8 Horn R A, Johnson C R. Matrix Analysis. New York: Cambridge University Press, 1985
- 9 Huang D S, Bao Z. Influences of shaped parameter α on recognition performance of Gauss radial basis function neural network. In: Chinese Conference on Neural Network, Xi'an, 1993, Xi'an: Xi'an Jiaotong University Press, 1993: 856~861

PATTERN RECOGNITION BY FUSION OF DIRECTIONAL BASIS FUNCTION AND PROBABILISTIC NEURAL NETWORK

Luo Xiongbiao Chen Tiequn Wan Ying
(College of Mechanical Engineering, South China University of Technology, Guangzhou 510640)

Abstract: The basal model of directional basis probabilistic neural network (DBPNN) and the corresponding algorithm and theory applied in pattern recognition are investigated aiming at utilizing the advantage and overcoming the shortcomings of directional basis function neural network (DBFNN) and probabilistic neural network (PNN). Its application to pattern recognition is from the results obtained in classification of cracks and porosity in weld defect. It can be seen that DBPNN has greater improvement in computation speed and classification, compared with the DBFNN and PNN.

Key words: Directional basis function neural network
Probabilistic neural network Fusion
Pattern recognition
Directional basis probabilistic neural network

作者简介: 罗雄彪, 男, 1979 年出生, 硕士研究生。主要从事超声检测缺陷模式识别和信号处理的研究。

E-mail: lxbxl_0207@sohu.com