

文章编号: 1009-038X(2000)06-0626-05

一种改进的自适应模式识别系统

葛洪伟, 朱建鸿

(无锡轻工大学信息与控制工程学院, 江苏无锡 214036)

摘要: 自适应模式识别系统具有大规模并行分布处理能力、通用性和自适应性, 因而显示了巨大的潜力. 但由于其结构的原因, 系统性能受到了一定的限制. 利用 P. Kanerva 提出的稀疏分布存贮原理对自适应模式识别系统进行改进, 提出了一种新的系统模型, 并就新系统的工作过程和主要特点作了较为详尽的叙述.

关键词: 模式识别; 稀疏分布存贮; 神经网络; 自适应

中图分类号: TP391.41

文献标识码: A

A Kind of Improved Adaptive Pattern Recognition System

GE Hong-wei, ZHU Jian-hong

(School of Information and Control Engineering, Wuxi University of Light Industry, Wuxi 214036)

Abstract: Adaptive pattern recognition system (WISARD) has found extensive applications in various areas because of its massive parallel distributed processing ability, versatility and self-adaptability. However, it has the weakness in the system structure. In this paper, a WISARD system was improved with the principle of the sparse distributed memory (SDM), proposed by P. Kanerva. The main feature and work process of the new system has been described in detail.

Key words: pattern recognition; sparse distributed memory; neural network; self-adaptability

自适应模式识别系统(WISARD)是英国 I. Aleksander 教授等人经过 10 余年的不懈努力于 1985 年研制成功的, 它采用了 RAM 网络实现的极度并行结构, 使系统能以 25 帧/s 的速率处理电视分辨率图象. 另外系统的实现与问题无关, 它不需要预先定义的面向问题的算法, 而只是通过对模式数据样本的直接学习而获得识别的知识, 因此系统具有良好的通用性和自适应性. 该通用系统推出并商品化后, 引起了学术界和工业界的很大兴趣, 并在工业视觉、字符识别、安全监视等诸多领域充分显示了它的应用潜能. 然而这种系统由于其结构的

原因, 存在着固有的缺陷, 比如: 在大维数模式识别问题中, 一般 n 元要求较大, 系统所需的 RAM 单元太多; 由于同类模式数据分布的凝聚性, 使 RAM 单元的利用率不高; 不能区分频繁出现和较少出现的模式, 降低了模式识别能力等. P. Kanerva 于 1988 年提出了稀疏分布存贮(SDM)的概念模型^[1], 它的基本思想是: 当存贮器地址空间很大时, 要构造相应的存贮器是不实际的, 此时可构造一个稀疏存贮器, 采用分布访问方式, 即当 2^n 个地址中的某个地址寻址存贮器时, 与该地址相似的若干稀疏存贮单元同时被访问, 并进行相应的读写工作. 作者借助

收稿日期: 2000-08-24; 修订日期: 2000-09-20.

作者简介: 葛洪伟(1967-), 男, 上海人, 工学硕士, 讲师.

SDM 思想,用稀疏存贮器替代 WISARD 的常规 RAM 网络,提出了一种既能保持 WISARD 原有优越性,又能克服其固有缺陷的新的系统模型.本文通过简例较为详细地介绍了新系统的训练和识别过程,并简要分析了其性能的改进和提高.

1 WISARD 系统模型简介

WISARD 的逻辑结构如图 1 所示.输入模式一般是 $m_1 \times m_2$ 的二值数据阵列,该阵列可以是一幅图象、一个字符或其它类型的模式数据.它被随机或确定地划分成 k 个子模式,每个子模式为 n 维,故称为 n 元子模式.每个 n 元分别连接到具有 2^n 个存贮单元的 RAM 地址端.这样, $k = m_1 \times m_2 / n$ 个内容寻址的 RAM 部件就构成了一个分类器的存贮空间.对于 T 类问题一般需要 T 个分类器与之对应.它们的训练单独进行,而分类则同时进行.

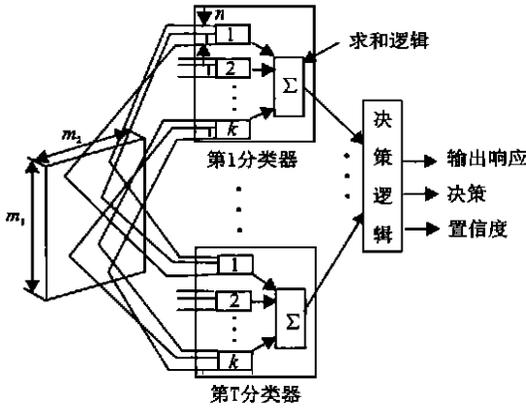


图 1 WISARD 的逻辑结构

Fig. 1 The logical structure of WISARD

在网络训练阶段,各分类器 RAM 置为“写”状态,当训练样本的 n 元模式数据寻址 RAM 时,被选中单元置为“1”.当一个模式类数据训练完毕后,再以同样的方式进行其它分类器的训练.在分类阶段,各分类器 RAM 置为“读”状态,当未知模式输入系统时,各 n 元子模式同时寻址各分类器的 RAM,相应单元的内容被读出,每个分类器的所有 RAM 输出之和形成该分类器的响应.决策逻辑根据各分类器的响应按最大响应决策原理对未知模式进行分类.有关 WISARD 系统更为详细的介绍,请参见文献[2,3].

2 稀疏分布式联想记忆模型

P. Kanerva 在文献[1]中提出了稀疏分布式联想记忆模型——SDM,其实质是 3 层神经网络,见图 2,其中第一与第二层神经元间的连接权值是固

定的,即采用预置的方法产生,它可用矩阵 A 表示,而第二、三层神经元间的连接权值阵 C 则通过推广 Hebb 学习规则而建立,见图 3.

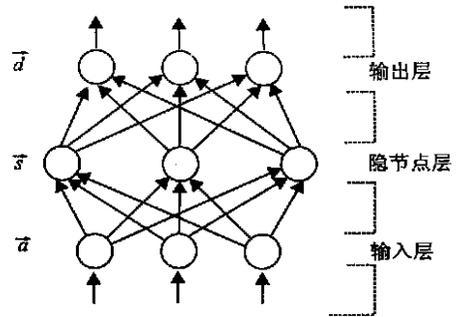


图 2 SDM 的多层结构

Fig. 2 The multilayer structure of SDM

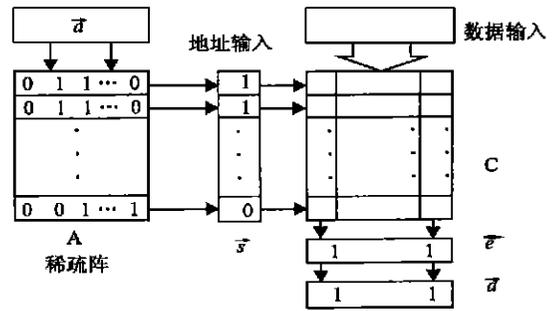


图 3 SDM 模型

Fig. 3 The model of SDM

令输入层的节点数为 n ,则 n 维输入地址 a 可构成 2^n 个地址的地址空间,当 n 较大时,该地址空间很难实现.为解决该问题,SDM 采用“稀疏存贮”的方法,即从 2^n 个地址中随机选取 m 个地址 $A_1, A_2 \dots A_m$ 构成稀疏地址阵 A ,而网络的学习过程如下:

设有 M 个(地址、数据)模式对,即训练集为 $\{(\vec{a}^1), \vec{d}^1, (\vec{a}^2), \vec{d}^2, \dots, (\vec{a}^M), \vec{d}^M)\}$,其中 $\vec{a}^i \in \{0, 1\}^n, \vec{d}^i \in \{-1, 1\}^n (1 \leq i \leq M)$,将该训练集作为输入,与 A 中预置内容逐个比较,选取 A 中落在以 \vec{a}^i 为中心、海明距离为 D 的超球 $V(\vec{a}^i, D)$ 内的地址作为数据 \vec{d}^i 的存贮地址,即若 A 中的某一地址恰好落在 $V(\vec{a}^i, D)$ 中,则对应单元输出 1,否则输出为 0,从而形成一个新的 m 维矢量 \vec{s} ,其形成过程可描述为:

$$\vec{s}^i = (s_1^i, s_2^i, \dots, s_m^i)^T$$

$$\text{其中 } s_j^i = \begin{cases} 0 & \text{若 } \rho(\vec{a}^i, A_j) > D \quad (1 \leq j \leq m) \\ 1 & \text{若 } \rho(\vec{a}^i, A_j) \leq D \end{cases}$$

$\rho(X, Y)$ 表示 X, Y 间的海明距离.

在向量 \vec{s} 形成的同时,数据 \vec{d}^i 写入由 \vec{s} 的“1”分量所指的 C 存贮器相应单元中.显然, C 中数据是稀疏分布地存贮的,当遇到 C 中的存贮单元重复写入时采用按位加再写入原则,此过程直至 M 个模式的学习完毕为止,从而建立了第二层到第三层的连接权值阵 C ,即 $C = \sum_{i=1}^M \vec{s}^i \vec{d}^i$.

当网络训练了 M 个模式对后,其联想过程如此进行:对任一给定的地址 \vec{a} ,仿学习过程以 \vec{a} 作为输入,从 A 中选取落在超球 $V(\vec{a}, D)$ 中的地址,然后形成向量 \vec{s} ,并将 \vec{s} 中“1”分量所指向的 C 中存贮单元内容按位加到向量 \vec{e} 中,经取阈值得到一输出数据 \vec{d} ,即 $\vec{d} = g(\vec{h})$,其中 $\vec{h} = \vec{s}^T C$, g 为某一阈值函数.若 \vec{a} 与 \vec{a}^i 近似,则 \vec{d} 与 \vec{d}^i 相似,这样就完成了联想记忆.

3 基于 SDM 思想的改进系统

WISARD 系统有许多优良特性,也有自身的弱点:

1) 在 WISARD 系统中 n 是一个重要参数.一般而言, n 元取得越大,模式类间的可分性就越好.对于诸如手写字符、动态脸谱等非确定性模式数据,常要求每类型 100 个甚至数百个训练模式以提高训练集的代表程度,这时为了避免因推广集过大而引起的分类错误,以及 RAM 存贮器的“饱和”问

题(即置“1”的 RAM 单元过多),就必须将 n 取得较大.但通过分析,我们可以得出系统所需容量为: $V_1 = m_1 \times m_2 / (n \cdot 2^n T)$,它将随 n 呈指数上升,系统成本增加很快.

2) 在实际应用中,真正代表模式数据性质的子模式总是频繁出现(称之为强子模式),而因噪声、畸变出现的子模式总是偶尔出现(称之为弱子模式).但 WISARD 系统并未记录子模式出现的频率,从而也就无法分辨强子模式和弱子模式,限制了系统的识别能力.

借助 SDM 的思想对系统加以改进,则可较好地弥补上述缺陷.

3.1 改进后的系统结构

改进后的系统结构如图 4 所示,和 WISARD 一样,输入模式被映射成 k 个子模式,每个子模式为 n 维,这些 n 元子模式分别与子矩阵 $A_i (i=1, 2, \dots, k)$ 相连, A_i 的输出连接 S_i ,而 S_i 则和 C_i 相连,其中 A_i, S_i, C_i 分别和 SDM 中的 A, S, C 矩阵结构相同.在新系统中, A_i 具有 m 个单元, n 位字长, $A_1, A_2 \dots A_k$ 构成了矩阵 A . S_i 字长为 1 位, $S_1, S_2 \dots S_k$ 构成了矩阵 S , C_i 为 b 位字长(b 的大小取决于训练集的大小,一般不会超过一个字节), $C_1, C_2 \dots C_k$ 构成 C 矩阵.和原系统一样,对于 T 类问题,则有 T 个分类器,每个分类器均有 A, S, C 3 个矩阵和一个求和 Σ 部件构成.分类器的训练单独进行,而在识别阶段, T 个分类器同时工作, T 个求和结果经决策逻辑得出相应识别结果.

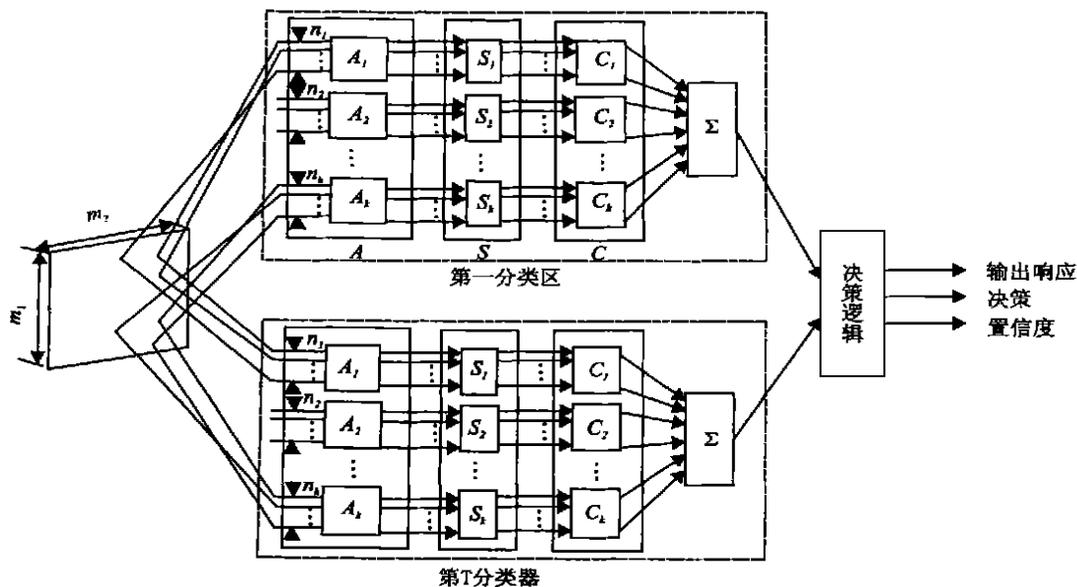


图 4 改进后的系统结构

Fig. 4 Improved system structure

在上述结构中,输入模式到 A 矩阵的映射基于 WISARD 的 N 元原理,而 A 、 S 、 C 矩阵的构造和连接则遵从 SDM 概念.不难看出,该系统实际上是 WISARD 的 N 元原理和 SDM 概念思想相结合的产物,当 A_i 的地址单元数为 2^n ,海明距离 $D=0$ 时,系统退化为 WISARD 系统,所以新系统可以看成是原 WISARD 系统的推广.

3.2 训练、识别过程

为叙述方便,现以一简化的两类分类器为例来

说明.例中输入模式大小为 3×3 位,输入模式和矩阵 A_i ($i=1, 2, 3$) 以相同的线性映射方式连接, n 元子模式的形式为 $n_i = \{x_i, y_i, z_i\}$, $i=1, 2, 3$. A_i 矩阵为 4×3 矩阵,即每个子矩阵有 4 个地址的存储空间,它们均被预置为 000、010、101、111,如图 5 所示. C 矩阵的各单元均预置为 0. 训练模式如图 6 的 T_1 、 T_2 、 T_3 所示,测试模式为图 6 中的 $T_4 \sim T_{11}$. 海明距离 $D=1$.

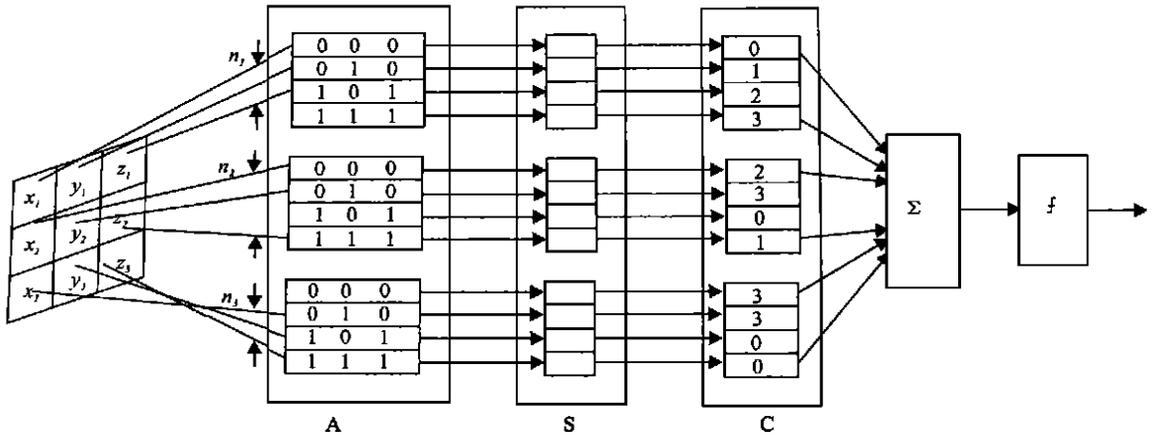


图 5 简化的两类分类器

Fig. 5 Simplified two-class classifier

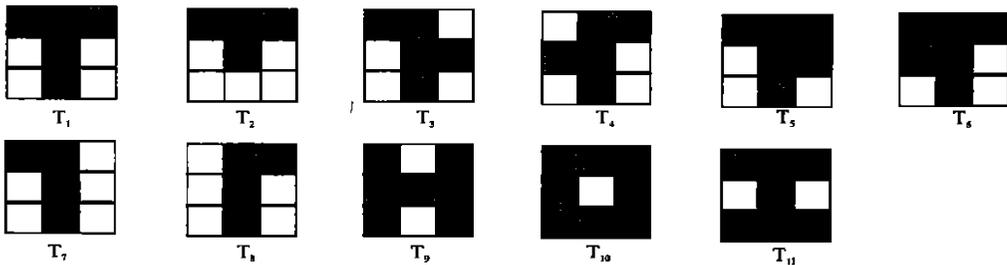


图 6 训练和测试样本

Fig. 6 Training and testing patterns

在训练阶段, C 矩阵的 RAM 为计数状态. 当训练 T_1 时, $n_1 = \{1, 1, 1\}$, 由于 $\rho(101, 111) \leq 1$, $\rho(111, 111) \leq 1$, 故 S_1 被置为 $(0, 0, 1, 1)^T$, C_1 被相应地写成 $(0, 0, 1, 1)^T$. 同样, $n_2 = \{0, 1, 0\}$, S_2 被置为 $(1, 1, 0, 0)^T$, C_2 被置为 $(1, 1, 0, 0)^T$. $n_3 = \{0, 1, 0\}$, C_3 也被写成 $(1, 1, 0, 0)^T$. 当训练 T_2 时, S_1 仍被置为 $(0, 0, 1, 1)^T$, C_1 中被选中的单元加 1, 即成 $(0, 0, 2, 2)^T$. 同样, C_2 为 $(2, 2, 0, 0)^T$, C_3 为 $(2, 2, 0, 0)^T$. 当训练完毕时, $C_1 \sim C_3$ 分别为 $(0, 1, 2, 3)^T$ 、 $(2, 3, 0, 1)^T$ 、 $(3, 3, 0, 0)^T$, 如图 5 所示.

在测试阶段, C 为读状态, 当输入 T_1 时, C_1 的第 3、第 4 单元被选中, C_2 、 C_3 的第 1、第 2 单元也被选中, 被选中的单元经求和输出响应 $r_{T1} = 16$. 图

6 中所有样本的响应值见表 1.

根据表 1 的各模式的响应值, 很容易找到一个阈值将它们分成两类, 即 $T_1 \sim T_8$ 为一类, $T_9 \sim T_{11}$ 为另一类. 取得上述响应值是合理的, 事实上 T_1 、 T_2 、 T_3 为训练样本, 它们可认为是字符 'T' 和它的变形, 而 $T_4 \sim T_8$ 则和字符 'T' 相似, 它们是训练集的推广, 是识别系统智能的体现, 而 T_9 为字符 'H', T_{10} 为字符 'O', T_{11} 为字符 'P', 它们均不属于 O_1 .

上述简化模型仅是为了说明系统的工作过程而设计的, 而在实际应用中, 系统将工作在模式维数较大的情况下, 只有这时才能充分体现系统的优越性.

表1 新系统的测试结果

Tab. 1 The testing results of new system

样本	响应	分类
T_1	16	
T_2	16	
T_3	14	
T_4	14	
T_5	15	O_1
T_6	15	
T_7	15	
T_8	15	
T_9	6	
T_{10}	6	O_2
T_{11}	10	

3.3 性能改进

从系统结构图1和图4的比较中,可以看出新系统仍将保持原WISARD系统的优良特性,如大规模并行处理、实时性、通用性、自适应性等,并且其性能将有如下改进.

1) 新系统所需的物理存贮器要少得多,其计算式为: $V_2 = m_1 \times m_2 / (n \cdot m \cdot (n+1+b) \cdot T)$. 其中 m 为稀疏地址矩阵的单元数, b 为每个分类器RAM的字长, T 为分类器的个数. 因为在实际应用中各分类器的 A 矩阵一般选取相同的随机地址, S 矩阵相应单元的置位情况也相同,所以 A 和 S 矩阵可以为各分类器所公用,此时 $V_2 = m_1 \times m_2 / (n \cdot m \cdot ((n+1)+b \cdot T))$. 而当分类器数目很大时, $V_2 = m_1 \times m_2 / (n \cdot m \cdot b \cdot T)$, 它随 m 线性增长. 实际应用中 n 较大, 此时 $m \ll 2^n$, $V_2 \ll V_1$. 例如: 当 $n=16$, $m=64$, $b=8$ 时, V_2 为 V_1 的 $1/128$.

2) 新系统能记忆子模式出现的频率, 因此能有效地区分强子模式和弱子模式, 从而进一步增强整个系统的模式识别能力. 这一点通过下面的例子可得到初步的证实(所做的工作均在仿真软件包上完

成).

① 简单示例

对上述训练、识别例子, 使用原WISARD系统识别, 在训练模式、 n 元大小等条件均不变的情况下, 各测试模式的响应情况如下表2所示:

表2 WISARD系统的测试结果

Tab. 2 The testing results of WISARD

样本	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8	T_9	T_{10}	T_{11}
响应	3	3	3	1	3	2	3	2	0	1	2

表2中两类模式的响应值较为相近, 判别逻辑难以作出正确的判别. 而在新系统中, 两类模式的响应值反差较大, 其敏感性和识别能力明显优于原系统.

② 手写字母识别实例

本实验仅针对小写的26个英文字母, 每个字母均有20个训练样本, 即共有20套520个样本, 这些样本分别由20个人工整书写获得, 样本被规格化成 24×24 的点阵. 在新系统中取 $n=12$, $m=64$, $D=6$, 而在原系统中取 $n=8$, 此时两个系统的识别率分别达到最佳. 当直接使用训练样本作为测试样本时, 新、旧系统的识别率分别为97.4%和93.1%, 而使用另外提供的5套工整书写的样本测试时, 则识别率分别为89.1%和81.3%.

4 结 语

作者利用SDM的概念和思想, 改进了WISARD系统, 大大提高了原系统的记忆容量和识别能力, 使系统能有效地分辨强子模式和弱子模式, 更加符合人脑的记忆、识别特性. 目前基于新模型的仿真软件包正用于印刷体、手写体汉字及实时图象的识别工作, 并已取得了初步的成果. 有关该模型严格的理论分析和应用研究还有待进一步深入.

参考文献

- [1] KANERVA P. Sparse Distributed Memory[M]. Cambridge: MA MIT Press, 1988.
- [2] ALEKSANDER I, THOMAS W V, BOWDEN P A. WISARD a radical step forward in image recognition[J]. *Sensor Review*, 1984, 4(3): 120~124.
- [3] ALEKSANDER I, MORTON H. An Introduction to neural computing[J]. *Chapman and Hall* 1992, 11(2): 70~90.

(责任编辑: 秦和平 李春丽)