

文章编号: 1003-501X(2011)06-0030-08

三角形剖分以及 PSO-BP 神经网络 在星图识别中的应用

张少迪^{1,2}, 王延杰¹, 孙宏海¹

(1. 中国科学院长春光学精密机械与物理研究所, 长春 130033;

2. 中国科学院研究生院, 北京 100049)

摘要: 为了实现星敏感器对航天器当前姿态的准确测量, 如何提高星图识别算法的实时性和鲁棒性成为星敏感器的关键技术。对星图识别过程中应用的模式提取、训练样本集的建立以及神经网络训练方式的改进等算法进行研究。首先, 设计一种基于星图特征的三角形剖分方法, 将视场内的恒星以三角形的方式组合起来, 提取星图模式, 建立完备的训练样本集, 使星图特征具有平移和旋转不变性。然后, 采用 BP 神经网络识别星图特征, 以权值矩阵代替导航星库, 一旦网络训练完成, 可以很快获得当前星图信息, 实现星敏感器星图识别算法的实时性和鲁棒性; 为了优化 BP 神经网络改进其自身缺点, 采用 PSO(粒子群算法)训练 BP 神经网络, 获取使 BP 神经网络趋近全局最优的初始权值和阈值, 使其加快收敛至全局最优。由实验结果表明, 该星图识别算法识别率达 100%。

关键词: 星图识别; 三角形剖分; BP 神经网络; PSO

中图分类号: TP391.41

文献标志码: A

doi: 10.3969/j.issn.1003-501X.2011.06.006

Triangulation and PSO-BP Neural Network Used in Star Pattern Recognition

ZHANG Shao-di^{1,2}, WANG Yan-jie¹, SUN Hong-hai¹

(1. Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China;

2. Graduate University of Chinese Academy of Sciences, Beijing 100049, China)

Abstract: In order to realize accurate measurement of aircraft's current attitude, how to improve real time and robustness of star pattern recognition is the key of star sensor. The algorithms for star pattern abstraction, training sample set creation and network training improvement are proposed. First, a method of triangulation based on the character of star image is designed to combine all the stars of current field of view, which is used to extract star pattern and create complete training samples. The character of star pattern extracted has the advantages of translation and rotation invariance. Then BP Neural Network serves to recognize the star pattern with the weight matrix instead of navigation library. It is very fast to acquire current star information when the network has finished training. Particle Swarm Optimization (PSO) serves to train BP Neural Network, which helps BP network converge to the most optimum value. The experimental results show that the success rate of accurate recognition is 100%.

Key words: star pattern recognition; triangulation; BP neural network; PSO

0 引 言

航天器在飞行过程中需要实时校正自身的飞行姿态, 星敏感器^[1]由于其体积小、重量轻、精度高、无

收稿日期: 2011-02-18; 收到修改稿日期: 2011-04-02

基金项目: 国家 863 高技术项目(2006AA703405F)

作者简介: 张少迪(1984-), 男(汉族), 吉林长春人。博士, 主要从事实时数字图像处理方面的研究。E-mail: zhangsd529@yahoo.com.cn。

误差累积等优点, 成为航天器首选的姿态敏感器。星敏感器为了解决“Lost-In-Space”问题, 采用自主星图识别算法^[2], 在没有任何先验姿态信息的指导下, 通过将当前视场内的观测星与其存储的导航星表进行对应匹配, 确定当前光轴指向, 完成姿态定位与调整。目前自主星图识别算法有很多种, 常见的有三角形算法、多边形算法、匹配组算法、KMP 算法等等。他们都有一个共同点, 都是通过将星图按照某个模式建模, 然后将导航星表衍生出若干星表, 再按照固定模式去查找星表完成匹配。虽然有若干改进算法提出, 但都避免不了重复查找星表, 导致实时性不高, 而且出现冗余信息等待筛选增加工作量。随着 BP 神经网络在模式识别领域内的应用日益广泛, 已有很多国内学者将其应用到星图识别。但是 BP 神经网络受初始权值影响大, 不仅收敛速度缓慢, 而且容易陷入局部极值, 在处理海量星图信息时必定受到限制。在本论文中针对 BP 神经网络的缺点, 提出一种适合星图特点的三角形剖分算法提取星图特征, 利用 PSO 训练神经网络使权值快速收敛到全局最优附近, 然后再用 BP 算法求出最佳权值。

1 三角形剖分算法

导航星在星表中的排列可以看做无序点集, 如果要描述导航星与其邻近导航星之间的拓扑连接关系, 就必须按照某种模式将他们组合起来。国内学者有很多利用三角形建模, 提出三角形算法以及在其基础上的凸多边形算法, 这主要是利用三角形是单纯形特点。但是传统三角形算法在识别过程中建立的三角形数目庞大而且冗余度高, 不但需要较大存储空间, 而且识别的压力也很大, 若不改进则很难在实际中应用。由于三角形剖分方法可以很好的描述点集中离散点之间的关系, 本文采用该方法建立星图描述模型。

1.1 平面点集三角形剖分的定义

定义 1: 给定平面内顶点集合 $\{V_i\}$, $i=1, 2, \dots, n$, 用不相交的直线段连接 V_i 与 V_j , $1 \leq i, j \leq n, i \neq j$ 使得 n 个点的凸壳内的每一个区域是一个三角形, 如图 1 所示, 这个过程称为三角剖分。剖分后形成的三角网格 T 也成为顶点集合 $\{V_i\}$ 的一个三角剖分。

一般来说, 对离散点集的三角剖分不是唯一的, 其中 Delaunay 三角剖分为最优^[3], 它满足最小内角最大准则。为了降低剖分算法复杂度并充分利用恒星的星等信息, 本文提出基于星图特征的三角形剖分算法。

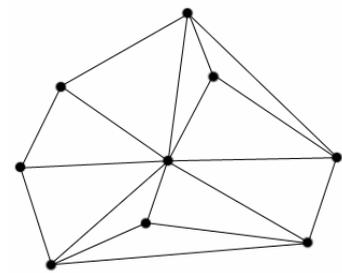


图 1 三角形剖分示意图

Fig.1 Triangulation figure

1.2 基于星图特征的三角形剖分算法

基于星图特征的三角形剖分(以下简称星三角剖分)步骤如下:

1) 将出现在星敏感器视场中的恒星按照星等由高到低排列, 并从 1 开始标记号码, 如果有两颗或两颗以上的恒星星等相同时, 则按照赤经坐标由小到大排序, 如图 2(a)所示;

2) 以 1 号星为端点顺次连接其他星, 如图 2(b);

3) 以 2 号星为端点, 顺次连接比它星号大的星, 由三角剖分的定义可知两两三角形之间三角形边不能相交, 因此在连接两颗星过程中, 如果与之前已经连接的线段相交, 则取消本次连接, 例如在连接线段 P_{2j} , $j > 2$ 的过程中, 要判断其与已经连接线端 P_{1h} , ($h > 1$ 且 $h \neq 2, h \neq j$) 是否相交, 其判断依据是两个不等式:

$$\begin{vmatrix} x_j - x_2 & y_j - y_2 \\ x_1 - x_2 & y_1 - y_2 \end{vmatrix} \begin{vmatrix} x_j - x_2 & y_j - y_2 \\ x_h - x_2 & y_h - y_2 \end{vmatrix} \leq 0 \quad (1)$$

$$\begin{vmatrix} x_h - x_1 & y_h - y_1 \\ x_2 - x_1 & y_2 - y_1 \end{vmatrix} \begin{vmatrix} x_h - x_1 & y_h - y_1 \\ x_j - x_1 & y_j - y_1 \end{vmatrix} \leq 0 \quad (2)$$

此恒星连接过程如图 2(c)所示;

4) 按照星号由小到大按照 3) 步描述的连接过程依次连接, 最后即可得到平面点集的三角剖分, 如图 2(d)所示。

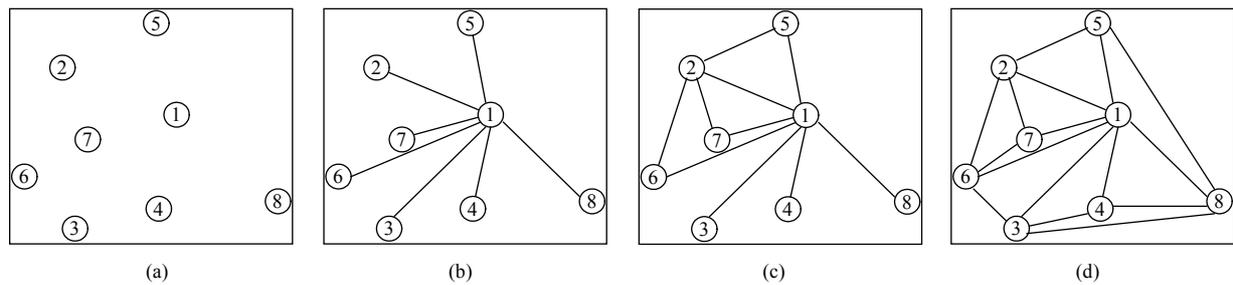


图2 基于星等的三角剖分示意图

Fig.2 Triangulation based on star magnitude

关于星三角剖分算法有几个实用结论,为了叙述方便首先给出几个定义:

定义 2: 视场内最亮的星即 1 号星为主星,其它星为副星;主星和副星之间的连接称为主距,副星之间的连接称为副距。

定义 3: 不同 n 颗星分布一般不同,应用星三角剖分得到的角距个数可能也不同,对于角距个数最多的 n 颗星分布,称为 n 颗星的最大分布;对于角距个数最少的 n 颗星分布,称为 n 颗星的最小分布。

由星三角剖分过程可以得到以下结论:

结论 1: 星三角剖分中每颗副星至少和其他一颗副星相连接。

证明: 反证法,如果一颗副星仅和主星相连接,则无法构成一个完整的三角形。

结论 2: 如果 $n-1$ 颗星最大分布的角距个数为 m ,则对于 n 颗星最大分布的角距个数为 $m+3$ 。

证明: 由于最大分布的 n 颗星凸壳呈三角形,如果再增加一颗暗星,最多增加一个主距和两个副距。

结论 3: 星三角剖分 n 颗星,构成三角形的角距个数最少为 $2n-3$,最多为 $3n-6$ 。

证明: 主距个数为 $n-1$ 个。对于已经剖分完毕的星图,再多加一颗星,为了讨论方便假设多加的星亮度比其它星都暗。下面利用数学归纳法证明:

首先证明角距个数最小值: 3 颗星时,角距个数为 $3=2\times 3-3$;多加一颗星即 4 颗星时,角距个数至少为 $5=2\times 4-3$;如果“星三角剖分 n 颗星,构成三角形的角距个数最少为 $2n-3$ ”结论成立,则 $n-1$ 颗星时角距个数最少为 $2\times(n-1)-3=2n-5$,如果多加一颗星即 n 颗星时,由结论 1 可知它至少和其他一颗副星相连,并且和主星相连,则角距个数为 $2n-5+2=2n-3$,由数学归纳法可知结论成立。

下面证明角距个数最大值: 3 颗星时,角距个数为 $3=3\times 3-6$;多加一颗星即 4 颗星时,角距个数最多为 $6=3\times 4-6$;如果“星三角剖分 n 颗星,构成三角形的角距个数最多为 $3n-6$ ”结论成立,则 $n-1$ 颗星时角距个数最多为 $3\times(n-1)-6=3n-9$,如果多加一颗星即 n 颗星时,由结论 2 可知 n 颗星时角距个数最多为 $3n-9+3=3n-6$,由数学归纳法可知结论成立。

由于出现在视场中的恒星分布随机,且考虑到神经网络输入层结构,在星三角剖分时会遇到如下问题:

1) 星图模式提取的方法很多,如提取星对角距、提取星三角形面积等。考虑到提取星三角形面积需要计算一对星对角距之间的夹角,容易造成误差累计,于是本文提取星对角距作为星图的模式特征。

2) 三角剖分以三角形为基础,即视场中至少要出现 3 颗星,如果视场中出现的恒星个数少于 3 个则不予识别。

3) 当视场中仅出现 3 颗星时,只能形成 1 个三角形即 3 个星对角距,由于神经网络的输入个数是一定的,如果神经网络考虑 3 颗星的情况,即神经网络的输入个数为 3,则每帧星图都提取 3 个星对角距。这样会造成星图信息大量丢失,导致神经网络的样本出现输入模式相同而学习目标不同的情况。本文实验表明,在大熊星座下随机出现在视场中的恒星个数平均为 6.7 颗星,由结论 3 可知,6 颗星三角剖分后的星对角距个数至少为 9 个,将神经网络的输入个数设置为 9,至于剖分星点后星对角距个数多于 9 个,则取长度最长的 9 个角距作为输入。对于角距小于 9 个的样本可以单独训练,其神经网络的输入个数为 3,至于剖分星点后星对角距个数多于 3 个而小于 9 个的,则取长度最长的 3 个角距作为输入。

4) 当视场中出现较多星点时,有些星点距离很近,先将距离很近(设一阈值)的星点归为一类,并只保

留该类中最亮星点。对于三颗星共线的情况, 由于取最长角距作为模式输入, 因此不影响三角形剖分。

由于飞行器的滚动, 造成所拍摄的星图围绕着星敏感器的视轴产生转动, 导致一些基于恒星在视场内坐标的星图模式提取算法对旋转过敏, 如图 3 所示。由于本文提出的星三角剖分算法与恒星在视场内的坐标无关, 只与剖分后提取的恒星之间角距有关, 因此只要恒星之间的相对位置不变, 所提取的星图模式就不会变化, 即本文提出的星三角剖分算法具有模式提取旋转、平移不变性。

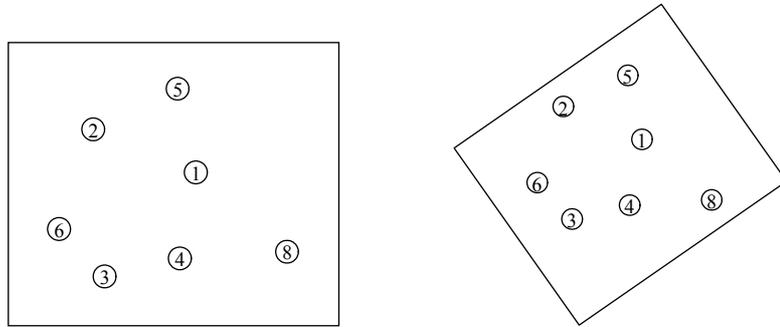


图 3 视场旋转对星图模式提取的影响

Fig.3 Impact on star pattern abstract for the reason of FOV rotation

2 基于 BP 神经网络的星图识别算法

神经网络是模仿人脑结构及功能的非线性信息处理系统, 具有大规模的并行计算与分布式存储能力, 且在处理信息的同时, 通过对信息的有监督和无监督学习, 实现对任意复杂函数的实值映射。神经网络应用到星图识别上, 其优点为: 模式的特征体现为各个神经元间权值的连接强度, 以权值矩阵代替了模式库, 不需要迭代就可以获得最终的识别结果, 可以大幅度地降低星载计算机系统的资源和星图匹配时间, 并且能提高图像匹配的识别率, 也可以改善目前识别系统的识别时间长、存储容量大的缺点。其中 BP 神经网络应用比较广泛, 下面简单介绍 BP(Back Propagation)算法。

2.1 BP 神经网络模型

理论证明一个三层 BP 神经网络(输入层、一个隐层和输出层)可以以任意精度逼近一个连续函数。假设输入层节点个数为 n , 隐层节点个数为 l , 输出层节点个数为 m , 则该前馈网络模型如图 4 所示。其中隐层节点个数的确定采用经验公式(3)确定范围, 再逐个实验取使神经网络性能最优的值, 经验公式为

$$l = \sqrt{n + m} + a \quad (3)$$

式中: n 为输入层神经元数, m 为输出层神经元数, l 为隐层神经元个数, a 为 1~10 之间的整数。

2.2 BP 算法的缺陷

三层 BP 神经网络算法由两部分组成: 信息的正向传递和误差的反向传输。在正向传递中, 输入信息从输入层经隐层逐层计算传向输出层, 每一层神经元的状态只影响下一层神经元的状态。如果在输出层没有得到期望的输出, 则计算输出层的误差变化值, 然后转向反向传播, 通过网络将误差信号沿原来的连接通路反传回来修改各层神经元的权值直至达到期望目标。BP 算法由于其通过一些学习规则来调整神经元之间的连接权值, 并通过非线性计算来识别目标, 具有思路清晰、状态稳定、结构严谨、实现容易等特点, 在模式识别等领域得到广泛应用。

传统的 BP 学习算法是一种随机梯度最小均方算法^[4-5], 每次迭代的梯度值受样本中噪声干扰的影响较

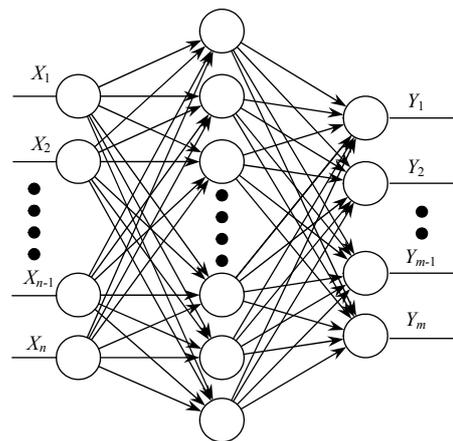


图 4 具有一个隐层的 BP 神经网络示意图

Fig.4 BP neuron network with one hide layer

大,一般使用批处理方法将多个样本的梯度进行平均以得到梯度估值。但是在训练模式样本数很大的情况下(例如本文星图模式的数量)这种方法势必增加每次迭代的计算量,并且这种平均作用将会忽略训练样本个体的差异性,使网络收敛速度慢和容易陷入局部极小值而无法继续训练等缺点,虽然采用自适应学习速率以及增加动量项改进方法,却仍没有改变 BP 算法的本质。本文引入 PSO 算法来优化 BP 神经网络。

2.3 利用 PSO 算法优化 BP 神经网络

在利用神经网络做星图识别的过程中,由于星点分布的离散性、星图模式的多样性以及初始权值和阈值的随机性,神经网络在训练过程中初始性能误差较大,而且有时会出现神经网络收敛到某个误差值而无法继续训练达到预期训练误差,即出现神经网络陷入局部极小值而偏离全局最优的情况。PSO(粒子群算法)为克服这一缺点提供了一个有效的途径^[7-9],本文利用 PSO 算法对 BP 神经网络进行初始权值训练,其基本思想是:将当前 BP 网络中所有的连接权值和神经元阈值的集合看作粒子群中的一个粒子,因此每个粒子的维数表示网络中起连接作用的权和神经元阈值的数量。以给定训练样本集的神经网络输出误差作为神经网络训练问题的适应函数,适应度值表示神经网络的误差阈值。误差越小则表明粒子在搜索中具有更好的性能,粒子在权值空间内移动搜索使得网络输出层的误差越小。标准 PSO 算法早期收敛速度较快,但到了寻优后期速度比较缓慢,这主要是因为算法收敛到局部极小值即早熟的原因,因此应该寻找保持其粒子多样性的算法来改进标准 PSO^[10-12]。

为了改进标准 PSO 存在的缺陷,在这里引入人工免疫系统算法中的定义:抗原、抗体、抗原和抗体之间的亲和度对应于目标函数、解、目标函数与解的适应度。在抗体初始化后,按照下面式(4)、式(5)进行迭代计算,首先选择亲和度高的抗体进行克隆操作,是抗体较多的聚集在“优”的位置,加快收敛速度;然后使用变异算子对克隆操作后的抗体进行变异操作,使抗体在“优”位置附近的各个方向进行搜索;最后使用基于浓度机制的多样性保持策略对其他抗体进行选择,以保证多样性。

搜寻 BP 神经网络最佳初始权值和阈值的实现过程为:

1) PSO 变量设置

设当前种群中的粒子 i 的位置为 BP 神经网络权值和阈值的集合 x_i : $X_i = [v_{i1} \dots v_{iN} \dots v_{iM} \dots w_{i1} \dots w_{iM} \dots w_{i1} \dots w_{iM}]^T$ 。当前粒子的维数为 $D = 1 \times N + m \times 1 + 1 + m$, 设当前粒子的飞翔速度为 V_i , 迄今为止搜索到的个体极值表示为 pb_i , 整个粒子群中所有粒子搜索到的全局极值表示为 gb , 则第 i 个粒子就是按照下面公式来更新自己速度和位置:

$$V_{id}^{n+1} = wV_{id}^n + C_1R_1(pb_{id}^n - X_{id}^n) + C_2R_2(gb_d^n - X_{id}^n) \quad (4)$$

$$X_{id}^{n+1} = X_{id}^n + V_{id}^{n+1} \quad (5)$$

其中 w 为惯性权重,较大的权重可以加强全局搜索能力,较小的权重能加强局部搜索能力。为了使算法具有较快的收敛速度,这里采用 Eberhart 的线性递减权重策略:

$$w(t) = (w_{ini} - w_{end})(T_{max} - t) / T_{max} + w_{end} \quad (6)$$

其中: t 为当前进化迭代次数, T_{max} 为最大迭代次数, w_{ini} 为初始惯性权重, w_{end} 为最大惯性权重。

2) PSO 改进

这里使用人工免疫系统中的克隆选择机制改进标准 PSO 算法。对于变异算子,采用增加随机扰动的方法使其能够脱离局部最优解,如式(7)所示:

$$gb_d^n = gb_d^n(1 + 0.5\eta) \quad (7)$$

其中 η 是服从 Gauss(0, 1)分布的随机变量。对于克隆变异操作后其他粒子的选择,采用基于浓度机制的多样性保持策略,定义第 i 个粒子的浓度为

$$D(x_i) = \frac{1}{\sum_{j=1}^{N+M} |f(x_i) - f(x_j)|}, \quad i = 1, 2, \dots, N + M \quad (8)$$

其中: M 为克隆粒子数, N 为粒子群大小, $f(x_i)$ 为第 i 个粒子的适应度。由式(8)可以得出基于粒子浓度的概率选择:

$$p(x_i) = \frac{\frac{1}{D(x_i)}}{\sum_{i=1}^{N+M} \frac{1}{D(x_i)}} = \frac{\sum_{j=1}^{N+M} |f(x_i) - f(x_j)|}{\sum_{i=1}^{N+M} \sum_{j=1}^{N+M} |f(x_i) - f(x_j)|}, \quad i = 1, 2, \dots, N + M \quad (9)$$

3) 评价函数与迭代终止条件

将更新后的粒子位置向量拆解为神经网络的权值和阈值并更新神经网络, 将神经网络的训练误差作为 PSO 的评价函数即适应度函数:

$$Ep = \frac{1}{2} \sum_{k=0}^{l-1} (d_k - Y_k)^2 \quad (10)$$

其中: d 、 Y 表示第样本输入时的期望输出和实际输出。当评价函数满足误差精度要求时或达到最大迭代次数时即可终止迭代循环。

3 实验与仿真

本文选用 SAO 星表, 恒星的赤经赤纬坐标采用 J2000 历元, 对大熊星座做星图识别。在该星座下, 星等高于 6 的恒星共有 114 颗, 以 $10^\circ \times 10^\circ$ 为视场大小搜索该星座, 首先对出现在视场中的星做三角剖分; 然后计算视场内剖分后相连接的恒星星对角距, 并按对角距的长度由大到小排列取前 9 个星对角距作为星图模式特征; 再取各个视场内星等最高的星的星号作为学习目标, 在这里基于 BP 神经网络的输出特点, 可以将这些目标导航星分类后用二进制表示; 最后利用 PSO 训练 BP 神经网络的算法训练神经网络。至于提取角距个数少于 9 个的视场按照 1.2 节的方法提取 3 个最长的对角距单独训练。在本文实验中, 由于两种样本集的建立和训练方式相同, 这里只讨论特征个数足够 9 个的样本集训练结果。

本文就一般的附加动量因子并自适应调整学习率算法和 PSO 选择初始权值算法分别给出训练误差曲线, 设定误差阈值为 0.01, 迭代次数上限为 7 000 次, 结果如图 5 所示。

从上图可以看出附加动量因子的自适应调整学习率算法经过 7 000 多次迭代训练仍没有使 BP 神经网络收敛至理想精度, 这是因为该算法随机选择初始权值, 如图所示性能曲线变化缓慢的区域说明网络陷入局部极小, 此时只有通过调整学习率使之脱离局部极值; 而性能曲线下落明显的区域说明此时网络正朝着梯度下降的方向进行, 即朝着全局最小的方向训练。但由于随机的初始权值离全局最小的距离过大, 导致网络训练不断的重复陷入局部极小、脱离局部极小的过程, 虽然最终能够达到全局最小, 但是需要长时间多迭代的训练。PSO 算法从全局着手进行搜索, 经过粒子速度与位置的更新, 能够搜索到靠近全局极小的区域, 为网络训练提供一个优秀的初始权值, 如图 6 所示 PSO 训练性能曲线初始的误差就要比前者小。虽然在之后的训练过程有可能陷入局部极小, 但由于此时已经很靠近全局最小, 经过学习率的自适应调整, 很快就可以到达全局最小, 如图所示性能曲线一直下降, 很少出现网络训练停滞不前的情况, 只需要 4 000 多次迭代即可达到理想精度。

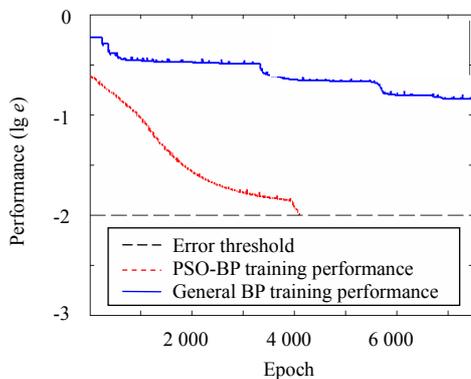


图 5 采用不同训练算法的神经网络训练误差曲线
Fig.5 Error curves of neuron network trained by different algorithm

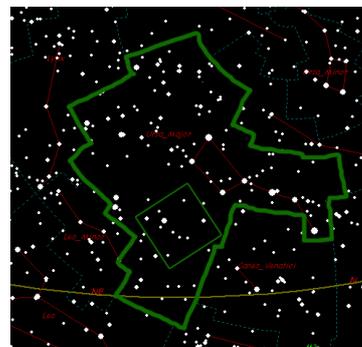


图 6 大熊星座模拟图
Fig.6 Simulative image of Ursa-major

本文在 HNSKY 2.3.0 模拟星空软件中, 将 CCD 框大小设置为 $10^{\circ} \times 10^{\circ}$, 在大熊星座范围内随机选取恒星作为观测值验证训练后的 BP 神经网络识别率, 具体步骤如下:

- 1) 设置 HNSKY 的极限星等为 6, 坐标采用 J2000 历元的赤经赤纬坐标系, 如图 6 所示, 用粗边框框起来的就是大熊星座, 小正方形代表随机视场;
- 2) 在大熊星座区域以预先设定的窗口大小选择恒星, 表 1 记录了三次实验结果;
- 3) 采用本文提出的三角剖分算法对出现在窗口中的星点进行剖分并提取对应的模式特征, 如图 7 所示;
- 4) 将提取的特征输入到训练好的神经网络进行识别, 最后给出试验结果, 如表 2 所示。

表 1 实验数据赤经赤纬坐标

Table 1 Stars right ascension and declination coordinate	
(RA, DEC) (J2000)	
1	(11h03m43.7s, 61°45'03");
	(10h51m23.7s, 69°19'12");
	(11h01m50.5s, 56°22'57");
	(10h51m11.1s, 56°34'56");
	(10h43m43.3s, 57°11'57");
	(10h35m9.7s, 57°04'58");
	(10h53m34.4s, 54°35'6");
2	(10h30m37.6s, 55°58'50")
	(12h29m57.3s, 58°24'21");
	(12h20m50.8s, 57°51'50");
	(12h25m03.2s, 56°46'40");
	(12h27m35.1s, 55°42'46");
	(12h15m25.6s, 57°01'57");
	(12h17m29.5s, 53°11'28");
3	(11h55m58.4s, 56°35'55");
	(11h53m49.8s, 53°41'41");
	(11h46m55.6s, 55°37'41");
	(11h35m04.9s, 54°47'07");
	(11h25m57.1s, 55°51'01")
	(10h43m04.1s, 69°04'34");
	(10h44m38.4s, 68°46'33");
(10h41m48.3s, 68°26'36");	
(10h21m03.4s, 68°44'51");	
(10h41m56.6s, 65°42'59");	
(10h24m07.9s, 65°33'59");	
(10h30m26.5s, 64°15'28")	

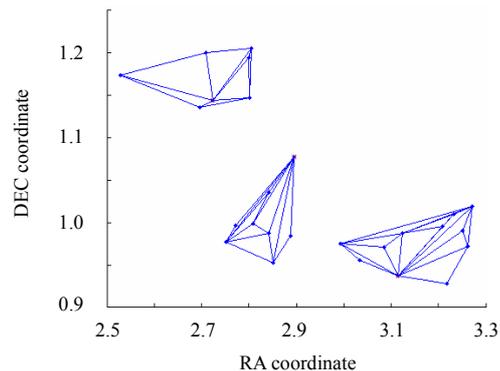


图 7 对应三个视场的三角形剖分结果

Fig.7 Triangulation results of three different field of view

表 2 试验结果

Table 2 Experimental results			
	Recognition time / ms	Target star	Success?
1	194	15 384	yes
2	187	28 179	yes
3	208	15 163	yes

本文实验是在实验计算机上进行的, 配置为: CPU 奔腾双核 E2220, 2G 内存, 并用 VC6.0 编程实现。表 2 所示试验结果, 其中识别时间是指用训练好的 BP 网络去识别星图所用的时间。虽然网络训练时间很长, 长达多于 5 min, 但由于训练数据基于 SAO 星表, 因此神经网络可以离线学习, 只需存储训练好的权值和阈值即可, 在识别过程中可以满足实时要求。

4 总 结

本文提出的基于星等的三角剖分算法与传统三角形算法在模式提取形式上相比较为简便, 如图 7 所示, 该方法不需要连接所有的三角形, 并且仅仅选择关键的角距建立模式样本集, 有效去除冗余信息。在网络训练方面, 利用 PSO 全局搜索能力, 预先搜索优秀初始权值, 并与随机初始权值做了比较, 从图 5 可以看出 PSO 训练方法在加快网络训练收敛方面作用突出。虽然 BP 网络的训练时间长, 迭代次数多, 但是其学习方式是线下学习, 一旦网络训练完毕, 存储网络权值等参数。由于星表是预先选定的, 因此每次识别之前无需重复训练网络, 直接进行识别即可, 从实验结果(表 2)可知满足实时性要求。

参考文献:

- [1] 钟红军. 流水线并行处理提高星敏感器数据更新率[J]. 光学 精密工程, 2009, 17(9): 2230-2235.

- ZHONG Hong-jun. Increasing update rate for star sensor by pipelining parallel processing method [J]. **Opt. Precision Eng**, 2009, **17**(9): 2230-2235.
- [2] 张磊. 快速全天自主星图识别[J]. 光学 精密工程, 2009, **17**(4): 909-915.
ZHANG Lei. Fast all-sky autonomous star identification [J]. **Opt. Precision Eng**, 2009, **17**(4): 909-915.
- [3] 周知. 三角剖分算法研究[D]. 哈尔滨: 哈尔滨理工大学, 2007.
ZHOU Zhi. Research on triangulation algorithm [D]. Harbin: Harbin University of Science and Technology, 2007.
- [4] 杨甲沛. 基于自适应学习速率的改进型 BP 算法研究[D]. 天津: 天津大学, 2008.
YANG Jia-pei. The research of improved BP algorithm based on self-adaptive learning rate [D]. Tianjin: Tianjin University, 2008.
- [5] 张蕾. 基于改进 BP 神经网络的物体识别研究[D]. 河南: 河南科技大学, 2008.
ZHANG Lei. The research on object recognition based on modified BP neural network [D]. Henan: Henan University of Technology and Engineering, 2008.
- [6] Ratnaweera A, Halgamuge S, Watson H. Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients [J]. **IEEE Transactions on Evolutionary Computation**(S1089-778X), 2004, **8**(3): 240-255.
- [7] 高海兵. 基于粒子群优化的神经网络训练算法研究[J]. 电子学报, 2004, **32**(9): 1572-1574.
GAO Hai-bing. Particle swarm optimization based algorithm for neural network learning [J]. **ACTA Electronic Sinica**, 2004, **32**(9): 1572-1574.
- [8] 高艳霞. 基于粒子群优化算法的 BP 神经网络在图像识别中的应用[J]. 武汉工业学院学报, 2006, **25**(4): 35-38.
GAO Yan-xia. Image recognition by BP neural networks model based on particle swarm optimization [J]. **Journal of Wuhan Polytechnic University**, 2006, **25**(4): 35-38.
- [9] 江涛. 一种改进的粒子群算法在 BP 网络中的应用研究[J]. 计算机科学, 2006, **33**(9): 164-166.
JIANG Tao. A study of application of an improved PSO algorithm in BP network [J]. **Computer Science**, 2006, **33**(9): 164-166.
- [10] 易云飞. 改进粒子群优化算法及其应用研究[D]. 广西: 中南民族大学, 2009.
YI Yun-fei. An improvement of particle swarm optimization algorithm and its application [D]. Guangxi: South-Central University for Nationalities, 2009.
- [11] 刘云龙, 林宝军, 艾勇. 基于改进粒子群优化算法的快速小目标检测[J]. 光电工程, 2009, **36**(10): 6-11.
LIU Yun-long, LIN Bao-jun, AI Yong. High Speed Small Target Detection Based on Improved Particle Swarm Optimization [J]. **Opto-Electronic Engineering**, 2009, **36**(10): 6-11.
- [12] Janson S, Middendorf M. A hierarchical particle swarm optimizer [C]// **Proceedings of IEEE Congress on Evolutionary Computation**. Canberra, Australia, Dec 8-12, 2003: 770-776.