

文章编号:1001-9081(2010)05-1293-04

一种增强型的粒子群优化算法

代军, 李国, 徐晨

(深圳大学 数学与计算科学学院, 广东 深圳 518060)

(water20007@163.com)

摘要:针对粒子群优化算法在进化后期容易陷入局部最优的缺点,提出了一种增强型的粒子群优化算法,即当粒子陷入局部极值点时,从增强粒子的自我学习能力,增强种群中其他相关粒子探索新区域的能力和增强粒子之间的信息交流三个方面来增强算法的寻优能力。数值实验结果表明,新算法具有很好的寻优性能。

关键词:粒子群优化;群体智能;惯性权重;压缩因子;局部极值;全局极值

中图分类号: TP18 **文献标志码:** A

Enhanced particle swarm optimization algorithm

DAI Jun, LI Guo, XU Chen

(College of Mathematics and Computational Science, Shenzhen University, Shenzhen Guangdong 518060, China)

Abstract: An Enhanced Particle Swarm Optimization (EPSO) algorithm was proposed to overcome the disadvantage of PSO such as easily falling into local optimal at the latter part of the evolution. In this algorithm, when particle fell into local extremum point, the algorithm enhanced its ability of searching global optimal value by enhancing the particle's self-study ability, the other relative particles' ability of exploring new search space and the information communication in particles. The experimental results indicate that the new method has good ability of searching optimal value.

Key words: Particle Swarm Optimization (PSO); swarm intelligence; inertia weight; constriction factor; local extremum; global extremum

0 引言

粒子群优化算法(Particle Swarm Optimization, PSO)是一种基于社会心理学模型的群智能优化算法^[1],与其他进化算法相比,该算法具有要设置的参数少,易于实现等优点,目前已被广泛应用于电力调度、目标优化等领域。尽管PSO具有明显的优势,但自身存在的最突出问题是:在进化后期容易陷入局部最优。对此,许多学者提出了改进方法,如文献[1]改进粒子的个体最优位置和全局极值的选取,提出了一种改进的PSO算法;文献[2]模拟天鹅迁徙的飞行过程,并根据天鹅在飞行过程中始终有领头鹅引领整个鹅群飞行的这一特点,提出了一种新型的PSO算法;文献[3]根据实验调查,限定惯性权重在0.4~0.6随机取值;文献[4]模拟学生的学习—考试机制,提出了一种学习—考试型的粒子群优化算法等。其中,文献[2,4]是通过模拟仿生现象提出来的,而PSO算法本身也是模拟鸟群觅食而提出来的,因此,通过对粒子群本身进行仿生分析是增强PSO算法性能的一个重要途径。本文从加强粒子自我学习和对新区域的探索,以及增加粒子之间的信息交流三个方面来改善PSO算法的寻优能力。

1 粒子群算法

1.1 粒子群算法的介绍

在粒子群算法中,每个粒子根据自己的飞行经验和种群中其他粒子的飞行经验来动态调整自己的位置^[4],具体如下。

设在 d 维搜索空间中,某一个种群有 s 个粒子,第 i 个粒子的第 t 代在该搜索空间中的位置为 $X_i^t = (x_{i1}^t, x_{i2}^t, \dots, x_{id}^t)$ ($i = 1, 2, \dots, s$),速度 $V_i^t = (v_{i1}^t, v_{i2}^t, \dots, v_{id}^t)$,第 i 个粒子在第 t 代的个体极值位置为 $P_i^t = (p_{i1}^t, p_{i2}^t, \dots, p_{id}^t)$,第 t 代的全局极值位置为 $P_g^t = (p_{g1}^t, p_{g2}^t, \dots, p_{gd}^t)$,每个粒子根据如下两个方程来更新自己:

$$V_i^{t+1} = V_i^t + c_1 \cdot \text{rand}() \cdot (P_i^t - X_i^t) + c_2 \cdot \text{rand}() \cdot (P_g^t - X_i^t) \quad (1)$$

$$X_i^{t+1} = X_i^t + V_i^{t+1} \quad (2)$$

其中: c_1, c_2 是学习因子,一般取值为2; $\text{rand}()$ 是 $[0, 1]$ 区间均匀分布的随机数。在式(1)右端项中, V_i^t 称为粒子的先前速度, $c_1 \cdot \text{rand}() \cdot (P_i^t - X_i^t)$ 为粒子的自我认知部分, $c_2 \cdot \text{rand}() \cdot (P_g^t - X_i^t)$ 为粒子的社会认知部分。粒子的先前速度与自我认知部分体现了粒子对自身经验的积累与信赖;粒子的社会认知部分体现了粒子之间的协作和信息交流。

PSO算法在初始化一群随机的粒子之后迭代运行,并且在每次迭代过程中,粒子通过个体极值位置(即 P_i^t)和全局极值位置(即 P_g^t)来更新自己,直到满足迭代停止条件为止。其算法流程如图1所示。

1.2 两种典型的粒子群优化算法

1.2.1 基于惯性权重的全局粒子群优化算法(GWPSO)

为了改善PSO算法的收敛精度,Shi等人^[5]首次在速度更新方程中引入惯性权重,即:

$$V_i^{t+1} = wV_i^t + c_1 \cdot \text{rand}() \cdot (P_i^t - X_i^t) + c_2 \cdot \text{rand}() \cdot (P_g^t - X_i^t) \quad (3)$$

收稿日期:2009-11-17;修回日期:2009-12-25。 基金项目:国家863计划项目(2006AA01A116);广东省自然科学基金资助项目(2008329);广东省省部产学研结合项目(2009B090300355)。

作者简介:代军(1985-),男,四川南充人,硕士研究生,主要研究方向:智能计算、小波分析;李国(1967-),男,河南信阳人,副教授,博士,主要研究方向:智能计算;徐晨(1965-),男,浙江海宁人,教授,博士生导师,主要研究方向:小波分析。

$$w = w_{\max} - \frac{iter}{iter_{\max}}(w_{\max} - w_{\min}) \quad (4)$$

其中: w 称为惯性权重, w_{\max} 为最大惯性权重, w_{\min} 为最小惯性权重; $iter$ 为当前迭代次数; $iter_{\max}$ 为总迭代次数。这样, w 就会随着迭代次数的增加而线性递减。对于全局搜索, 在进化初期, w 较大使得粒子拥有较高的探索能力; 而在进化后期, 较小的 w 使得粒子拥有较为精细的搜索能力, 以便收敛到更好的解。而且, Shi 等人的仿真实验也表明, w 线性递减取得了较好的实验效果。一般地, w 在 0.4~0.9 取值。

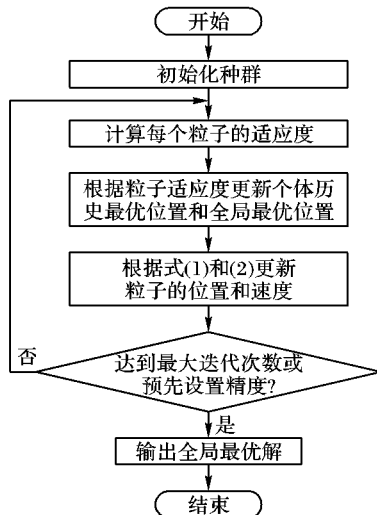


图1 PSO算法流程

1.2.2 基于压缩因子的全局粒子群优化算法(GCPSO)

为了确保 PSO 算法的收敛, Clerc^[6] 引入了压缩因子, 其速度更新公式如下:

$$V_i^{t+1} = \alpha \cdot (V_i^t + c_1 \cdot \text{rand}() \cdot (P_{i1}^t - X_i^t) + c_2 \cdot \text{rand}() \cdot (P_{i2}^t - X_i^t)) \quad (5)$$

其中 $\alpha = \frac{2}{|2 - \varphi - \sqrt{\varphi^2 - 4\varphi}|}$, $\varphi = c_1 + c_2 > 4$ 。一般情况下, $c_1 = c_2 = 2.05$, 所以, $\alpha = 0.729$ 。

2 一种增强型的粒子群优化算法

Shi 等人在对粒子群优化算法研究的时候发现, PSO 算法在初期搜索效率很高, 但在接近最优解时搜索速度降低, 并且很难得到精确最优解^[7]。这是因为 PSO 算法在后期陷入了局部最优, 产生局部最优的原因可以从如下两个方面来解释, 并试图从解释中找到增强 PSO 寻优性能的方法。

1) 从数值仿真实验来看, 粒子陷入局部最优表现为: 全局最优位置连续很多代变化很小或者没有变化, 这说明种群向全局最优区域的搜索能力丧失。原有的粒子更新方式已经丧失(或部分丧失)指导粒子进化的能力, 而作为主体的粒子, 此时应当在改变粒子更新方式的同时, 着重增强种群中粒子的自我学习能力。

2) 从社会心理学角度来看, 粒子陷入局部最优是因为: 对于种群中的某个粒子 i 来说, 随着粒子自身的不断进化与成长, 对信息的需求也逐渐增大, 单依赖于 P_i^t, P_g^t 这两个“极值”已经不能满足粒子对于信息的需求, 所以, 此时应当增强种群中其他粒子探索新区域的能力, 并将探索到的新信息及时与粒子 i 充分共享, 即增强粒子之间的信息交流。

根据以上分析, 提出了一种增强型的粒子群优化算法(EPSO), 其执行步骤如下。

步骤1 随机初始化粒子的位置、速度, 并计算粒子的适应值、粒子群的全局极值和粒子的个体极值。

步骤2 根据式(4)计算惯性权重, 然后根据式(3)和式(2)分别更新粒子的速度和位置。

步骤3 判断粒子是否陷入局部最优, 若是, 执行如下过程; 否则, 转向步骤4。

a) 随机选取种群中一半的粒子, 并初始化一个计数器 K , 称之为粒子的信息交流次数。

b) 寻找与随机选取的粒子 i 当前适应度差异最大的粒子 P_d^t 。

c) 将粒子 i 的位置按式(6)修改, 之后, 计算粒子在新位置的适应度。

$$x_{ij}^t = c_1 \cdot x_{ij}^t + c_2 \cdot \text{rand}() \cdot (P_{ir_1}^t + \text{rand}()) + c_2 \cdot \text{rand}() \cdot (P_{gr_2}^t + \text{rand}()) + c_2 \cdot \text{rand}() \cdot (P_{dr_3}^t + \text{rand}()) \quad (6)$$

d) 将步骤 b)、c) 顺序执行 K 次, 然后转向步骤4。

步骤4 若算法终止条件满足, 输出相关参数并停机; 否则, 迭代次数加1, 转向步骤2。

式(6)中, c_1, c_2 是学习因子, 如前所述, 取值为2; $\text{rand}()$ 是 $[0, 1]$ 区间均匀分布的随机数; r_1, r_2, r_3 是 $[1, d]$ 区间均匀分布的随机整数; $c_1 \cdot x_{ij}^t$ 是为了增强粒子的自我学习能力; $P_{ir_1}^t + \text{rand}(), P_{gr_2}^t + \text{rand}(), P_{dr_3}^t + \text{rand}()$ 是为了增强种群中相关粒子自我探索新区域的能力; $c_2 \cdot \text{rand}() \cdot (P_{dr_3}^t + \text{rand}())$ 和设置信息交流次数 K 是为了增强粒子之间的信息交流。另外, 式(6)实际上是式(3)的一个变式, 借助速度更新方程的结构, 可以有效共享种群的相关信息。

3 优化性能测试

3.1 标准测试函数

在实验中, 选取4个经典测试函数对 EPSO 算法和文献中其他 PSO 算法的寻优性能进行测试。所选4个测试函数都是极难优化的函数, 即使在维数很小的时候, 也难以收敛到全局最优解。

函数1 $f(X) = \sum_{i=1}^{n-1} 100(x_i^2 - x_{i+1})^2 + (1 - x_i)^2$;
 $-5.12 \leq x_i \leq 5.12$, 该函数在 $x_i = 1$ 时取得最小值0。

函数2 $f(X) = \sum_{i=1}^n [x_i^2 - 10\cos(2\pi x_i) + 10]$;
 $-15 \leq x_i \leq 15$, 该函数在 $x_i = 0$ 取得最小值0。

函数3 $f(X) = \sum_{i=1}^n -x_i \sin(\sqrt{|x_i|}) + 418.9829 \times n$;
 $-500 \leq x_i \leq 500$, 该函数在 $x_i = 420.9687$ 取得最小值0。

函数4 $f(X) = \sum_{i=1}^n \frac{1}{4000} x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$;
 $-600 \leq x_i \leq 600$, 该函数在 $x_i = 0$ 取得最小值0。

3.2 粒子信息交流次数的讨论

在 EPSO 算法中, 由于粒子加强自我学习和自我探索, 使得种群中产生了大量新的信息。

如何在粒子之间交流这些信息是改善 PSO 算法性能的关键, 而粒子信息交流次数决定了粒子之间信息交流的充分程度, 因此, 深入探讨粒子信息交流次数与算法优化性能之间的关系显得尤为必要。为此, 对以上4个函数进行了如下

实验。

首先,将粒子信息交流次数 K 限定在一个较小的范围之内,因为太大的信息交流次数会增加运算时间,维数较大时尤其明显。所以在实验中, K 限定取 $1 \sim 12$ 的整数;其次,在 K 值给定之后 ($K = 1$ 开始),每个函数运行 4 次(注:每次运行 30 个结果,相关参数设置见表 1),得到函数全局平均极值(求 \lg)与信息交流次数之间的关系如图 2 所示。

表 1 实验参数设置

函数	维数	种群大小	搜索区间	预设精度
函数 1	60	80	$[-5.12, 5.12]$	≤ 1
函数 2	60	80	$[-15, 15]$	≤ 1
函数 3	60	80	$[-500, 500]$	≤ 0.01
函数 4	60	80	$[-600, 600]$	≤ 0.01

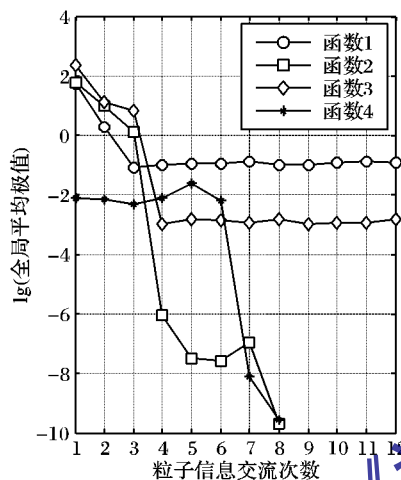


图 2 EPSO 与信息交流次数关系

由图 2 可以看出, K 在一定的范围内,函数 1、3 随着 K 的增加而增强;随着 K 的继续增加,算法则稳定下来;多峰函数 2、4 则随着 K 的增加而有所起伏,但最终也稳定下来,特别是在 $K = 9 \sim 12$ 时,函数 2、4 均取得全局最优值 0 (由于是取对数,所以不能画出图)。产生这种现象的原因是:在最初几次交流的时候,种群中由于加入了新的信息而增强了种群的多样性,使得算法向更优区域搜索前进;但当这类信息持续增加时,种群中的多样性受到抑制,使得算法停滞不前,但算法的寻优性能已得到明显提高。由于每个函数的性态有很大的差异,如函数 1 非凸、病态,函数 2 多峰等,使得每个函数对信息的需求也不相同,但 $K = 9$ 时,函数 2、4 刚好取得了全局最优值,同时函数 1、3 精度稳定,所以在实验中,取 $K = 9$ 。

3.3 与其他 PSO 算法的对比

为比较增强型 PSO 算法的寻优性能,将 EPSO 与典型算法 GWPSO 和 GCP SO,以及文献[1]中的改进的粒子群优化算法(本文记为 MPSO)和文献[3]的中基于随机惯量权重的快速粒子群优化算法(本文记为 RWPSO)进行了对比实验。实验参数设置见表 1,每个函数迭代 2000 次,每个函数运行 30 次,以平均值作为最终结果。其中,EPSO 算法判断粒子陷入局部最优的条件是:全局极值连续 U 次没有变化,其中 U 为 $[15, 20]$ 区间的随机整数。其他算法的参数设置按相关参考文献的要求设置。实验结果见表 2,其中,如果函数在指定迭代次数内没有达到预设精度,则函数该次的收敛代数数为指定迭代次数;否则,第一次满足预设精度的迭代次数记为该函数收敛代数;平均收敛代数是函数收敛代数总和与实验次数

的比值;收敛率则是满足预设精度的实验次数与实验总次数的比值。各个函数的进化曲线如图 3~6 所示(为显示粒子进化细节,图 3 未取对数,图 4~6 均求 \lg)。

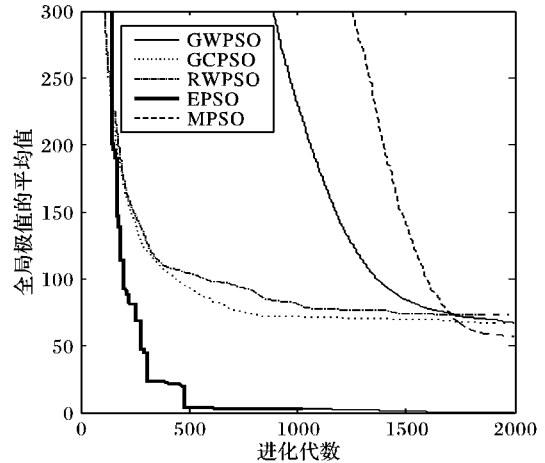


图 3 函数 1 进化过程

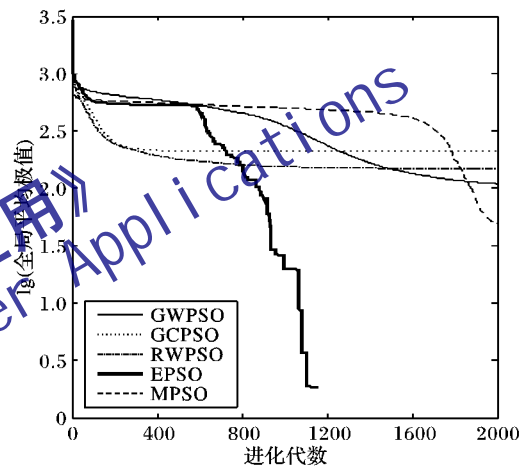


图 4 函数 2 进化过程

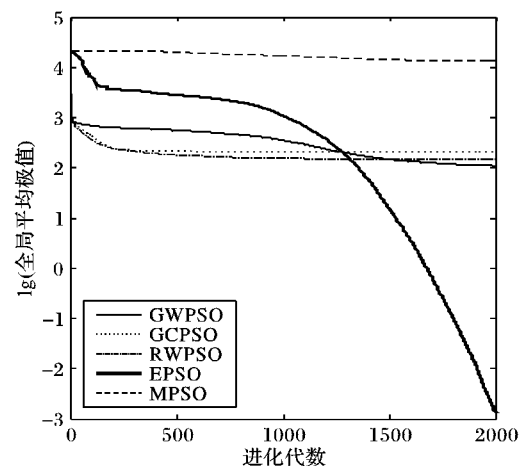


图 5 函数 3 进化过程

由表 2 及图 3~6 可以看出,EPSO 与其他四种 PSO 算法相比,无论是在精度、收敛率还是收敛速度方面,都优于其他四种算法,这主要是因为,在 EPSO 中,算法一旦陷入局部极值,立刻可以获得种群中的其他大量信息,并充分交流共享,从而增强了种群的多样性,使得算法向更优的区域继续搜索前进。

4 结语

在粒子群的进化过程中,粒子极易遇到局部极值点,本

文首先对其原因进行了分析,并从增强粒子的自我学习,探索和充分交流信息三个方面来增强算法的优化能力。同时,本文做了较为深入的实验,探讨了粒子信息交流次数与算法优

化能力之间的关系,并将由此得到的结果来指导对比实验。在对四个很难优化函数的仿真中,EPSO 均表现出良好的寻优性能,增强了 PSO 算法的优化能力。

表2 各种算法性能的比较结果

函数	算法	平均值	最优值	方差	平均收敛代数	收敛率/%
函数 1	GWPSO	$6.744\,227 \times 10^1$	$5.137\,062 \times 10^1$	$4.951\,475 \times 10^2$	2000.0	0
	GCPSO	$6.683\,114 \times 10^1$	$2.430\,301 \times 10^1$	$8.140\,681 \times 10^2$	2000.0	0
	RWPSO	$7.314\,423 \times 10^1$	$5.322\,714 \times 10^1$	$6.354\,027 \times 10^2$	2000.0	0
	EPSO	$1.015\,209 \times 10^{-1}$	$2.286\,708 \times 10^{-2}$	$4.789\,041 \times 10^{-3}$	1560.6	100
	MPSO	$5.709\,732 \times 10^1$	$5.531\,710 \times 10^1$	$4.553\,048 \times 10^{-1}$	2000.0	0
函数 2	GWPSO	$1.093\,513 \times 10^2$	$6.965\,213 \times 10^1$	$4.875\,738 \times 10^2$	2000.0	0
	GCPSO	$2.115\,075 \times 10^2$	$1.422\,788 \times 10^2$	$1.597\,943 \times 10^3$	2000.0	0
	RWPSO	$1.487\,703 \times 10^2$	$8.258\,149 \times 10^1$	$8.489\,987 \times 10^2$	2000.0	0
	EPSO	0	0	0	780.7	100
	MPSO	$4.905\,362 \times 10^1$	$3.381\,412 \times 10^1$	$1.136\,660 \times 10^2$	2000.0	0
函数 3	GWPSO	$1.146\,357 \times 10^4$	$8.491\,332 \times 10^3$	$1.418\,522 \times 10^6$	2000.0	0
	GCPSO	$1.251\,675 \times 10^4$	$7.935\,953 \times 10^3$	$3.090\,487 \times 10^6$	2000.0	0
	RWPSO	$1.250\,773 \times 10^4$	$1.007\,061 \times 10^4$	$1.713\,230 \times 10^6$	2000.0	0
	EPSO	$1.313\,760 \times 10^{-3}$	$2.457\,067 \times 10^{-4}$	$1.054\,660 \times 10^{-6}$	1898.4	100
	MPSO	$1.340\,485 \times 10^4$	$8.963\,729 \times 10^3$	$5.833\,282 \times 10^6$	2000.0	0
函数 4	GWPSO	$5.992\,100 \times 10^{-3}$	$1.057\,228 \times 10^{-4}$	$7.809\,730 \times 10^{-5}$	1892.6	6.7
	GCPSO	$5.082\,946 \times 10^{-2}$	$2.656\,295 \times 10^{-18}$	$8.470\,611 \times 10^{-3}$	1335.9	46.7
	RWPSO	$1.310\,599 \times 10^{-2}$	$3.252\,607 \times 10^{-19}$	$2.955\,787 \times 10^{-4}$	1244.8	53.3
	EPSO	0	0	0	334.2	100.0
	MPSO	$1.416\,459 \times 10^{-1}$	$4.890\,940 \times 10^{-3}$	$7.431\,590 \times 10^{-2}$	1999.2	3.3

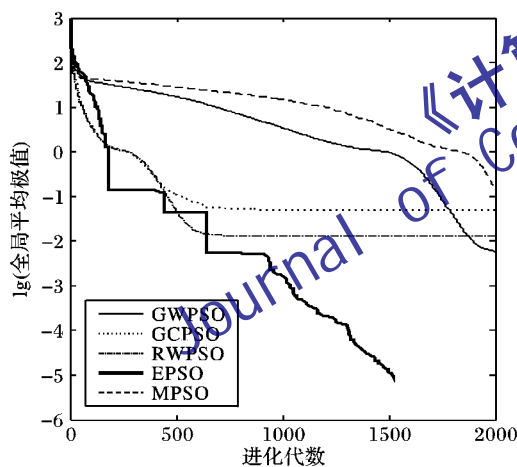


图6 函数4 进化过程

参考文献:

[1] 王晓英,邢志栋,黄瑞平.改进的粒子群优化算法[J].计算机应

用与软件,2008,25(5):84-86.

[2] WANG JUN - WEI, WANG DING - WEI. Particle swarm optimization with a leader and followers [J]. Progress in Natural Science, 2008, 18(11): 1437-1443.

[3] 黄轩,张军,詹志辉.基于随机惯性权重的快速粒子群优化算法[J].计算机工程与设计,2009,30(3):647-650.

[4] 代军,李国,徐晨.学习—考试型的粒子群优化算法[J].计算机工程与应用,2009,45(18):48-50.

[5] SHI Y, EBERHART R C. A modified particle swarm optimizer [C]// Proceedings of the IEEE International Conference on Evolutionary Computation. Washington, DC: IEEE Press, 1998: 69-73.

[6] CLERC M. The swarm and queen: Towards a deterministic and adaptive particle swarm optimization [C]// Proceedings of Congress on Evolutionary Computation. Washington, DC: IEEE Press, 1999: 1951-1957.

[7] 余炳辉.粒子群优化算法试验研究及扩展[D].武汉:华中科技大学,2007.

(上接第1267页)

[3] JAIN R. Difference and accumulative difference pictures in dynamic scene analysis [J]. Image and Vision Computing, 1984, 2(2): 99-108.

[4] CHEN BAISHENG, LEI YUNQI. Indoor and outdoor people detection and shadow suppression by exploiting HSV color information [C]// The Fourth International Conference on Computer and Information Technology. Washington DC: IEEE Computer Society, 2004: 137-142.

[5] WU B-F, JUANG J-H, TSAI P-T, et al. A new vehicle detection approach in traffic jam conditions [C]// CIISP 2007: IEEE Symposium on Computational Intelligence in Image and Signal Processing,

Washington DC: IEEE Press, 2007: 1-6.

[6] 潘薇,游志胜,吴鹏,等.基于模糊聚类和卡尔曼滤波的运动目标检测[J].计算机应用,2005,25(1):123-124.

[7] 赵明,李娜,陈纯.采用统计推断的自动视频对象分割[J].计算机辅助设计与图形学学报,2005,15(3):318-323.

[8] 李强.一种新的自适应背景建立与更新算法[J].电子科技,2007(10):6-8.

[9] 胡燕,嵇启春,李智杰,等.视频监控中背景的提取和更新算法[J].计算技术与自动化,2009,28(1):142-144.

[10] 向海霞,王卫星.车辆检测中的自适应背景更新算法研究[J].计算机工程与应用,2009,45(11):235-237.