

# 具有记忆性的自适应布谷鸟搜索算法

秦 岭,戴睿闻

(南京工业大学 计算机科学与技术学院,江苏 南京 211800)

**摘 要:** 针对标准布谷鸟搜索算法后期收敛速度慢、易陷入局部最优等缺点,本文提出了一种具有记忆性的自适应布谷鸟算法(MACS)。根据种群适应度与个体适应度准确判定算法收敛程度并分别对不同鸟窝进行自适应调整,使算法在迭代过程中保持收敛速度与精度的平衡;通过在偏好随机游动环节中引入记忆策略,算法能够充分利用上一代的有效信息,提高了其跳出局部最优的能力。仿真实验结果表明,MACS算法各方面性能较标准算法及其改进版本具有明显优势。

**关键词:** 布谷鸟算法;莱维飞行;收敛程度;自适应方法;记忆策略

**中图分类号:** TP18

**文献标识码:** A

**文章编号:** 1000-7180(2017)03-0015-05

## Adaptive Cuckoo Search Algorithm with Memory Strategy

QIN Ling,DAI Rui-wen

(College of Computer Science & Technology, Nanjing Tech University, Nanjing 211800, China)

**Abstract:** The cuckoo search algorithm (CSA) is a new nature-inspired algorithm which is simple and efficient. This algorithm uses Lévy flights random walk and biased random walk to search for new solutions. Aiming at increasing the performance of cuckoo search algorithm via improving parameter, this paper presents Adaptive Cuckoo Search Algorithm with Memory Strategy (MACS). By adjusting Lévy flight adaptively based on premature convergence degree and the fitness of the cuckoo, the algorithm makes a compromise between global search and local search. The ability of local search has improved via introducing memory strategy. Simulation results have showed that the performance of MACS is obviously improved in the aspect of the convergence and accuracy compared with the CS algorithm and other improved editions.

**Key words:** cuckoo search algorithm; Lévy flight; average fitness; adaptive method; memory strategy

### 1 引言

布谷鸟搜索算法(CSA)<sup>[1]</sup>模拟了布谷鸟独特的寻窝产卵的行为,并引入了自然界鸟类、果蝇运动轨迹的莱维飞行机制<sup>[2]</sup>,使其能够快速有效地寻找到最优解。虽然布谷鸟搜索算法控制参数少、简单高效且易于实现,但是该算法同其他群体智能算法一样,也存在局部搜索能力弱、易陷入局部最优等缺点。目前通过改进算法参数、调整算法结构提高算法性能是一个重要的研究方向。国内外众多学者对此展开研究,并提出一些改进方法:文献[3]引入贪婪机制提高算法的收敛精度,然而时间开销增大难以避免;

文献[4]针对莱维随机步长提出一种自适应版本以加强全局搜索但未改进偏好随机游动环节,算法的局部能力提升不足;文献[5]在莱维飞行的更新公式将步长因子改为服从 $[0,1]$ 均匀分布的压缩因子,此方法虽然大幅提高了算法全局搜索能力但由于没有考虑对局部搜索的优化,易发生早熟收敛。文献[6]针对莱维飞行和发现概率引入了自适应方法增强了收敛性能。然而改进算法步长调节范围较小,在迭代前期不能充分遍历寻优空间,在求解单峰函数时效果不佳。

本文分析了布谷鸟搜索算法的寻优过程,根据群体收敛程度和个体适应度对莱维飞行进行自适应

调整,使种群在迭代过程中保持收敛速度与精度的平衡;通过在偏好随机游动环节中引入记忆策略,算法能够充分利用上一代的有效信息,提高了其跳出局部最优的能力。

## 2 布谷鸟搜索算法

布谷鸟繁衍后代的方式独特且具有攻击性,它们将自己的卵偷偷产入宿主的巢穴并由宿主负责孵化。在某些情况下,宿主也会发现巢穴中的陌生卵,此时宿主鸟将抛出这个卵或者遗弃该巢穴,并选择在其他地方重新筑巢。为了简化和模拟布谷鸟的繁殖方式,Yang 和 Deb 假定 3 条规则<sup>[7]</sup>:

规则 1:每只布谷鸟一次只产一个卵,并把该卵放在随机选择的鸟巢中,该鸟窝也被称为寄身巢。

规则 2:具有最高质量的卵的寄生巢将被保留到下一代。

规则 3:可用鸟巢的数量是固定的,而且鸟巢中寄身卵被发现的概率是  $p_a$ ,且  $p_a=0.25$ 。

在标准算法中,每个鸟窝表示 1 个可行解,算法目标是利用新的潜在的较优解替换鸟巢中较差的解<sup>[8]</sup>。

算法首先对种群中每个解采用莱维飞行产生新的解。按照适应度大小保留较好的解。鸟窝的更新公式如下:

$$X_{g+1,i} = X_{g,i} + \alpha \oplus \text{Levy}(\beta) \quad (1)$$

式中, $X_{g,i}$  表示第  $g$  代第  $i$  个解; $\alpha$  是步长因子,用于控制随机搜索的范围。为点对点乘法。该式描述的随机游动是一个马尔可夫链,即下一代的位置仅取决于当前的位置。为从当前最优解获取步长信息,采用下式计算步长因子:

$$\alpha = \alpha_0 (X_{g,i} - X_{\text{best}}) \quad (2)$$

式中, $\alpha_0$  为常数<sup>[8]</sup>, $X_{\text{best}}$  表示当前最优解。

Yang 等人对莱维分布函数进行了简化,得到幂次形式的概率密度函数:

$$\text{Levy}(\beta) \sim \mu = t^{-\beta}, 1 < \beta < 3 \quad (3)$$

由式(3)可知,布谷鸟连续的位置变化形成了一个带有重尾的概率分布,从本质上体现了布谷鸟飞行的过程。

为了便于计算,文献[5]采用式(4)计算莱维随机数:

$$\text{Levy}(\beta) \sim \frac{\varphi \times \mu}{|\nu|^{1/\beta}} \quad (4)$$

式中, $\mu, \nu$  服从标准正态分布, $\beta = 1.5$ <sup>[9]</sup>。

$$\varphi = \left( \frac{\tau(1+\beta) \times \sin(\pi \times \beta/2)}{\tau((\frac{1+\beta}{2}) \times \beta \times 2^{(\beta-1)/2})} \right)^{1/\beta} \quad (5)$$

显然,结合式(1)~(5),在莱维飞行环节中 CS 算法采用公式(6)生成新的解  $X_i$ :

$$X_{g+1,i} = X_{g,i} + \alpha_0 \frac{\varphi \times \mu}{|\nu|^{1/\beta}} (X_{g,i} - X_{\text{best}}) \quad (6)$$

算法在按发现概率丢弃部分解后,采用偏好随机游动重新生成相同数量的新解。评价新解的适应度后,输出质量最高的解完成一次迭代。偏好随机游动如式(7)所示:

$$X_{g+1,i} = X_{g,i} + r(X_{g,j} - X_{g,k}) \quad (7)$$

式中, $r$  是压缩因子,为一个在(0,1)区间上的均匀分布随机数; $X_{g,j}$  和  $X_{g,k}$  表示第  $g$  代的两个随机解。

## 3 具有记忆性的自适应布谷鸟搜索算法

莱维飞行控制着布谷鸟算法的全局搜索<sup>[8]</sup>。步长因子的选取影响着布谷鸟算法的搜索特性。文献[4-6]以及实验研究表明,莱维步长与算法的收敛速度呈正相关与计算精度呈负相关。基于此性质,多种自适应策略应用于标准算法的改进中。

### 3.1 收敛程度判定

自适应方法的引入使得种群在迭代过程中具备适当的参数并朝最优位置进化,提高了算法的性能。

布谷鸟搜索算法的众多改进版本提出了多种自适应方案,将寻优步长与迭代次数构成的负相关函数作为步长更新公式,这些自适应算法改善了标准算法因步长因子固定从而收敛速度慢、易陷入局部最优的缺点。然而此类方法的缺陷亦显而易见:算法在处理复杂优化问题时难以准确把握迭代次数与收敛程度的关系,不同优化问题的收敛程度不一样,仅通过迭代次数的自适应方法通用性不佳。

算法对于自身所处阶段的判定在很大程度上影响自适应的性能。布谷鸟算法首先以式(6)产生新解,个体在追踪当前最优位置的过程中表现出趋同性,导致解更新的停滞,使得算法的种群多样性逐渐丧失,发生早熟收敛。若降低趋同性,算法的收敛速度必然受到影响。为准确把握种群的收敛程度,本文采用解的平均适应度以评价算法的收敛情况。具体操作如下:

假设种群为  $n$ ,第  $i$  代鸟窝  $k$  的适应度值为  $f_k$ ,

则种群平均适应度  $f_{\text{avg}} = \frac{1}{n} \sum_{k=1}^n f_k$ ;  $f_m$  为适应度最优的解,令  $f_{\text{avg}}$  为  $f_{\text{avg}}$  与  $f_m$  间的平均值。当接近收敛

时,种群必然在最优位置附近,各鸟窝的适应度将减小,因此解的平均适应度能够反映算法的寻优情况.故定义程度因子  $\Delta = f_{\text{avg}} - f_m$  作为判定收敛程度的标准.程度因子越小表示解的差异性减小,算法越趋于早熟收敛.

### 3.2 自适应莱维飞行

根据 3.1 节提出的收敛程度判定方法,本文将种群中各鸟窝分为三类:收敛性较好的鸟窝、一般鸟窝与发散性较强的鸟窝.每类鸟窝各具特点,所以在进化中采用相同的自适应方法,无法平衡全局搜索与局部搜索,影响了算法的收敛性能.故针对每种鸟窝进行合理的自适应调整.

三类鸟窝的自适应策略为:

$$f_m < f_k < f_{\text{avg}}$$

这类鸟窝的种群中质量较优的解,已接近全局最优位置,可称为精英鸟窝.此时应以较小的莱维步长因子指导其获取高精度的全局最优解.其公式如下所示:

$$\alpha_1 = \alpha_0 \cdot \left| \frac{f_k - f_{\text{avg}}}{f_{\text{min}} - f_{\text{avg}}} \right| \quad (8)$$

式(8)可得适应度越小的鸟窝即精英鸟窝会以越小的莱维步长进行精确搜索,从而加强了算法的局部开发能力.

$$f_{\text{avg}} < f_k < f_{\text{avg}}$$

这类鸟窝的适应度介于程度因子与平均适应度之间,可称为一般鸟窝.该类个体能够较好的平衡全局搜索和局部搜索,故以标准布谷鸟算法的步长因子进行解的更新.

$$f_k > f_{\text{avg}}$$

这类鸟窝的适应度大于种群的平均适应度,具有较强的发散性,可称为劣等鸟窝.由于莱维飞行具有高度随机性,且布谷鸟算法每轮迭代只保留一个最优解,故此类鸟窝有机会转化为其他类型的个体.此时设置莱维步长因子应在寻优过程中具有自适应性.改进公式如下所示:

$$\alpha_2 = 1.5 - \frac{1}{1 + \exp(-\Delta)} \quad (9)$$

若鸟窝较为集中,则  $\Delta$  较小,此时步长增大,加强全局搜索,使种群跳出局部最优;若鸟窝分布较为分散,则  $\Delta$  较大,此时步长减小,强化局部搜索,使种群趋于收敛.

考虑到三类鸟窝在寻优过程中存在相互转化的可能,将精英鸟窝与劣等鸟窝的步长更新公式与种群的收敛程度相关联,使用程度因子准确反映算法的收敛

程度并对其全局搜索与局部搜索做出了很好的折衷.

### 3.3 记忆鸟窝

布谷鸟搜索算法的另一环节是偏好随机游动.式(7)采用双随机解间的偏移量( $X_{g,j} - X_{g,k}$ )与压缩因子  $r$  乘积作为随机游动步长,按发现概率进一步产生新的解.这种双随机解的形式并未充分整合已知解的有利信息,无法保证更新解的质量.本文采用记忆策略,用当前代的已知解替换式(7)中的一个随机解,提高算法的局部搜索能力.改进式如下式(10).

$$X_{g+1,i} = X_{g,i} + r(X_{g,i} - X_{g,k}) \quad (10)$$

### 3.4 算法流程

综上所述,MACS 算法执行步骤可做如下归纳:

Step1:初始化鸟巢(解),设置算法各项参数,计算所有鸟巢(解)的适应度.

Step2:根据第 1 步的计算,对收敛程度进行判定,将鸟窝分为 3 类,并分别按 3 种自适应策略对解进行更新.

Step3:计算更新解的适应度,若新的解适应度更高,则替换旧的解.

Step4:按发现概率随机淘汰部分解,并采用具有记忆性的偏好随机游动(公式 10)产生与淘汰解数量相同的新解.

Step5:计算更新的鸟巢适应度值,输出当前最优解.

Step6:判断算法终止条件,若满足则获得结果,否则重复步骤 Step2~Step6.

## 4 仿真分析

### 4.1 实验准备

为证明 MACS 具有更快的收敛速度和搜索精度,选取六个涵盖单峰及多峰问题的测试函数对基本 CS 算法、文献[10]的 ICS 算法、文献[6]的 NICS 算法及本文所提 MACS 算法进行对比实验. ICS 和 NICS 算法皆是引入自适应莱维飞行的改进算法,具有代表性.实验中各算法独立运行 30 次,性能指标包括最小、最大适应度、平均适应度及标准差,如表 1 所示.

### 4.2 仿真结果与分析

实验结果如表 2 及图 1~图 6 所示. 仿真图像中横坐标为迭代次数,纵坐标为目标函数适应度的数量级,本文所提算法的收敛曲线以"o"表示.由表 2 及各仿真进化曲线图可见 MACS 算法较标准 CS 算法及两个自适应改进算法在收敛精度及收敛速度

表 1 测试函数及参数设定

| 测试函数            | 搜索空间                                                                                        | 理论最优             | 维度  |
|-----------------|---------------------------------------------------------------------------------------------|------------------|-----|
| Sphere          | $f_1(x)=\sum_{i=1}^N x_i^2$                                                                 | $[-100,100]$     | 030 |
| Schwefel P2. 22 | $f_2(x)=\sum_{i=1}^N  x_i  + \prod_{i=1}^N  x_i $                                           | $[-5,5]$         | 030 |
| Schaffer        | $f_3(x)=\frac{\sin^2 \sqrt{\sum_{i=1}^N x_i^2}-0.5}{[1+0.001(\sum_{i=1}^N  x_i ^2)]^2}+0.5$ | $[-10,10]$       | 030 |
| Rosenbrock      | $f_4(x)=\sum_{i=1}^{N-1} [100(x_{i+1}-x_i)^2+(x_i-1)^2]$                                    | $[-2.048,2.048]$ | 030 |
| Rastrigin       | $f_5(x)=\sum_{i=1}^N [x_i^2-10\cos(2\pi x_i)+10]$                                           | $[-5.12,5.12]$   | 030 |
| Griewank        | $f_6(x)=\frac{1}{4000}\sum_{i=1}^N x_i^2-\prod_{i=1}^N \cos(\frac{x_i}{\sqrt{i}})+1$        | $[-600,600]$     | 030 |

上具备优势. 以下结合实验结果具体说明.

表 2 四种优化算法寻优结果

| 函数              | 算法   | 最小适应度    | 最大适应度    | 平均适应度    | 标准差      |
|-----------------|------|----------|----------|----------|----------|
| Sphere          | CS   | 3.24E-4  | 2.62E-3  | 1.01E-3  | 4.07E-3  |
|                 | ICS  | 2.87E-8  | 7.61E-6  | 2.03E-6  | 3.55E-6  |
|                 | NICS | 3.12E-6  | 6.82E-5  | 2.48E-5  | 4.47E-6  |
|                 | MACS | 9.80E-15 | 3.82E-13 | 6.71E-14 | 1.67E-13 |
| Schwefel P2. 22 | CS   | 1.37E-1  | 7.54     | 4.83     | 2.47E-1  |
|                 | ICS  | 1.09E-3  | 3.65E-2  | 6.69E-3  | 3.51E-3  |
|                 | NICS | 5.97E-4  | 2.40E-3  | 1.06E-3  | 6.71E-2  |
|                 | MACS | 8.97E-11 | 3.27E-9  | 5.71E-10 | 4.15E-10 |
| Schaffer        | CS   | 0        | 0        | 0        | 0        |
|                 | ICS  | 0        | 0        | 0        | 0        |
|                 | NICS | 0        | 0        | 0        | 0        |
|                 | MACS | 0        | 0        | 0        | 0        |
| Rosenbrock      | CS   | 8.32E-1  | 6.18     | 4.74     | 2.81     |
|                 | ICS  | 6.77E-1  | 3.90     | 1.56     | 4.97E-1  |
|                 | NICS | 4.43E-1  | 1.29     | 7.92E-1  | 5.08E-1  |
|                 | MACS | 2.47E-4  | 6.99E-3  | 1.62E-3  | 3.74E-5  |
| Rastrigin       | CS   | 9.44     | 25.96    | 13.38    | 2.57     |
|                 | ICS  | 7.08E-2  | 3.43E-1  | 6.52E-1  | 5.64E-1  |
|                 | NICS | 3.55E-2  | 1.29E-1  | 4.14E-1  | 3.96E-1  |
|                 | MACS | 9.41E-6  | 5.57E-5  | 3.04E-6  | 2.64E-5  |
| Griewank        | CS   | 5.01E-2  | 7.72E-1  | 5.46E-1  | 3.98E-1  |
|                 | ICS  | 4.02E-5  | 5.17E-3  | 3.39E-4  | 4.29E-4  |
|                 | NICS | 1.53E-6  | 4.84E-4  | 2.70E-4  | 1.39E-5  |
|                 | MACS | 0        | 2.56E-8  | 5.18E-9  | 2.81E-9  |

均 Sphere 及 Schwefel P2. 22 为简单的单峰函数, 全局最优解易于求解, 常用来衡量算法的寻优速度. ICS 与 NICS 算法因对莱维步长调节范围较小, 求解精度提升不显著. MACS 算法寻优精度更高, 较其他三个算法提高了 10 个左右数量级且鲁棒性最强.

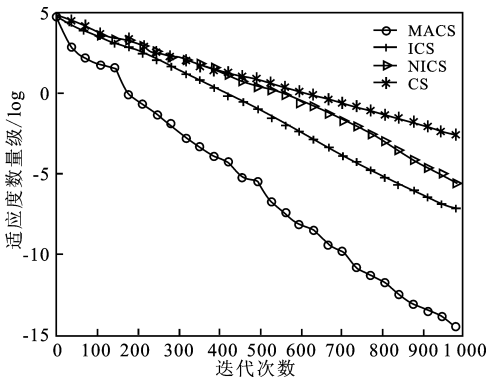


图 1 Sphere 进化曲线

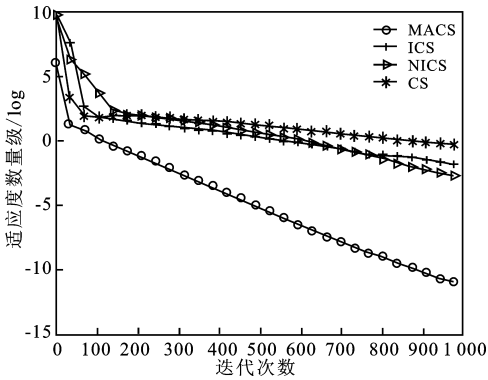


图 2 Schwefel P2. 22 进化曲线

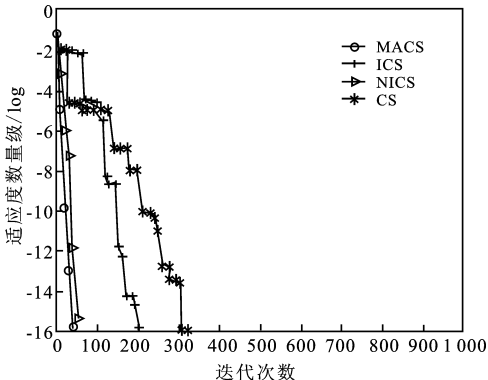


图 3 Schaffer 进化曲线

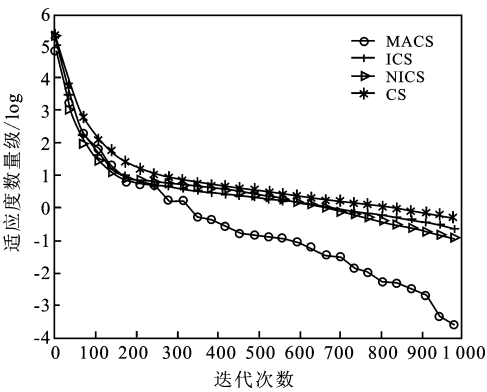


图 4 Rosenbrock 进化曲线

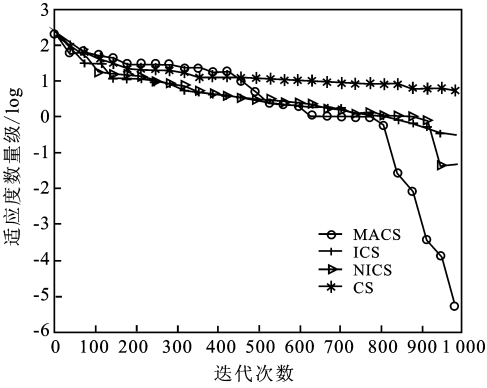


图 5 Rastrigin 进化曲线

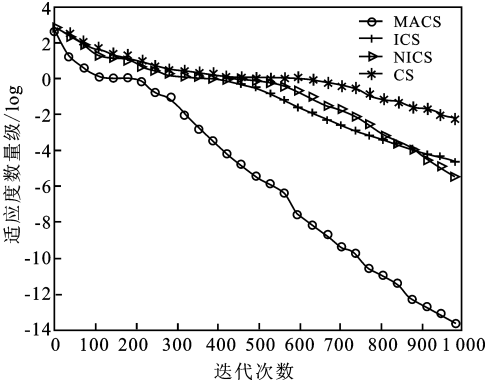


图 6 Griewank 进化曲线

Schaffer 函数是复杂单峰函数,四个算法在测试中均能收敛到理论最优位置,但本文所提算法的收敛速度更快.若运用在实时系统中以速度作为性能指标,MACS 算法将更具优势. Rosenbrock 函数也是复杂单峰函数,该函数的全局最优点位于一个平滑、狭长的抛物线形山谷内,优化算法难以找到搜索方向.虽然本文所提算法并未就寻优方向进行优化,但引入记忆鸟窝后,算法能够充分利用历史信息,加强了其局部搜索能力.由此可知,改进算法全局最优解、平均最优解及稳定性皆好于 ICS、NICS 及标准 CS 算法.

针对多峰函数的寻优情况,这两个测试函数是典型的多模态非线性全局优化函数,适合衡量算法的全局搜索能力. Rastrigin 函数为不可分离的典型多峰函数,具有强烈震荡且其局部最优位置随着正弦波动,一般算法难以接近全局最优值.实验结果可见,标准 CS 算法在 10 附近陷入局部最优;ICS 和 NICS 算法虽然采用自适应方法,但缺乏对种群收敛程度的准确判断,改进效果不佳.本文所提算法以程度因子指导算法向全局最优位置进化,记忆策略的引入加强了其局部搜索能力,算法成功跳出局部最优. Griewank 函数具有震荡性,且全局最优位置被大量局部最优解包围.由仿真结果可见,ICS 及 NICS 算法改进效果略好于标准算法.具有记忆性的改进算法引入收敛程度判定标准,算法的自适应操作更为准确,因而获得满意的结果.

由仿真实验可得,MACS 在处理单峰及多峰复杂问题时能够表现出更好的性能优势,改进效果显著.

5 结束语

本文分析了布谷鸟搜索算法的寻优过程与特性,提出了一种具有记忆性的自适应布谷鸟搜索算法(MACS).根据种群平均适应度与个体适应度准确判定算法收敛程度并分别对不同鸟窝进行自适应调整,使种群在迭代过程中保持收敛速度与精度的平衡;通过在偏好随机游动环节中引入记忆策略,算法能够充分利用上一代的有效信息,提高了其跳出局部最优的能力.仿真实验结果证明,改进的布谷鸟搜索算法具有较好的寻优能力及鲁棒性,克服了标准算法易陷入局部最优的缺点,在解决连续函数问题上具有优势.

参考文献:

[1] Yang X S. Nature-inspired metaheuristic algorithms [M]. UK: Luniver Press, 2008; 81-96.  
[2] Aziz Ouaraab, Belai Ahiod, Xinshe Yang. Random-key cuckoo search for the traveling salesman problem[J]. Soft Computing, 2015, 19 (4): 1099-1106. [doi: 10. 1007/s00500-014-1322-9].  
[3] 王李进,尹义龙. 逐维改进的布谷鸟搜索算法 [J]. 软件学报, 2013, 24(11): 2687-2698.  
[4] Walton S, Hassan O, Morgan K, et al. Modified cuckoo search: A new gradient free optimization algorithm[J]. Chaos, Solutions & Fractals, 2011, 44(9): 710-718.

- [6] He X C, Yung N H C. Curvature scale space corner detector with adaptive threshold and dynamic region of support[C]//International Conference on Pattern Recognition 2004. Hongkong, 2004: 791-794.
- [7] Rosten E, Drummond T. Machine learning for highspeed corner detection[C]// Computer Vision-ECCV2006. Berlin, Heidelberg, Springer, 2006: 430-443.
- [8] 安维胜, 余让明, 伍玉铃. 基于 FAST 和 SURF 的图像配准算法[J]. 计算机工程, 2015, 41(10): 232-235.
- [9] 梁艳菊, 李庆, 陈大鹏, 等. 一种快速鲁棒的 LOG-FAST 角点算法[J]. 计算机科学, 2012, 39(6): 251-254.
- [10] Yang T, Chen J X, Wei L I, et al. An adaptive susan corner detector based on lifting wavelet transform[J]. Microelectronics & Computer, 2009, 26(7): 156-159.
- [11] 徐树奎, 李国辉, 张军, 等. 基于多方向对称和匀质约束的 SUSAN 棋盘格角点检测算法[J]. 计算机科学, 2011, 37(9): 248-252.
- [12] 任雁. 角点检测方法研究[J]. 机械工程与自动化,

2009, 37(1): 198-200.

- [13] 薛金龙. 基于角点的图像特征提取与匹配算法研究[D]. 大连: 大连理工大学, 2014.
- [14] Huang H, Ma G, Zhuang Y. Vehicle license plate location based on harris corner detection[C]// Neural Networks, 2008. IJCNN 2008. (IEEE World Congress on Computational Intelligence). IEEE Interna. China, Hong Kong, 2008: 352-355.
- [15] 刘溪. 基于特征的图像配准算法与应用研究[D]. 北京: 北京理工大学, 2015.

### 作者简介:

刘 亮 男, (1993-), 硕士研究生. 研究方向为计算机视觉和图像处理. E-mail: 18262281433@163.com.

王 平 男, (1970-), 硕士, 教授. 研究方向为国防工程智能化.

孙 亮 男, (1991-), 博士研究生. 研究方向为计算机视觉和图像处理.

(上接第 19 页)

- [5] Wang Lijin, Yin Yilong, Zhong Yiwen. Cuckoo search with varied scaling factor[J]. Front. Comput, 2015, 9(4): 623-635.
- [6] Jin Qibing, Qi Linfeng. Novel improved cuckoo search for PID controller design[J]. Transactions of the Institute of Measurement and Control, 2014, 37(6): 1-11.
- [7] Yang X S, Deb S. cuckoo search: recent advances and applications[J]. Neural Computing and Applications, 2014, 24(1): 169-174.
- [8] Yang X S, Deb S. Cuckoo search via Levy flights[C]// Proceedings of World Congress on Nature & Biologically Inspired Computing, NaBIC 2009. Washington: IEEE Publications, 2009: 210-214.
- [9] Yang X S, Deb S. Engineering optimization by cuckoo

search[J]. Int'l Journal of Mathematical Modeling and Numerical Optimization, 2010, 1(4): 330-343. [doi: 10.1504/IJMMNO.2010.03543].

- [10] Valian E, Mohanna S, Tavakoli S. Improved cuckoo search algorithm for global optimization[J]. Int'l Journal of Communications and Information Technology, 2011, 1(1): 31-44.

### 作者简介:

秦 岭 男, (1980-), 硕士, 讲师. 研究方向为计算机应用与人工智能.

戴睿闻(通讯作者) 男, (1991-), 硕士研究生. 研究方向为人工智能与优化控制. E-mail: drw\_nj@163.com.