

基于果蝇优化算法改进的粒子滤波 及其在目标跟踪中的应用*

韩锲[†], 张赫

(中南大学 交通运输工程学院, 湖南 长沙 410075)

摘要:针对粒子滤波算法重采样导致的样本贫化问题,提出一种基于果蝇优化思想的粒子滤波算法.该方法视粒子权值为个体适应度值,并将果蝇不断从低浓度的地方飞向高浓度的地方的觅食寻优过程引入到粒子滤波当中,驱使粒子不断向高似然区域移动,提高了粒子群的整体质量.为了解决标准果蝇优化算法易陷入早熟的问题,将遗传算法中的交叉、变异操作自适应地应用到果蝇优化算法寻优过程当中.首先通过交叉操作改善粒子分布,当果蝇优化算法陷入局部最优时,再采用柯西变异扰动,促使算法快速跳出局部极值并继续搜索全局极值.通过非线性模型仿真以及目标跟踪实验表明该算法有效提高了非线性系统状态估计精度,具有较好的稳定性,同时降低了状态估计所需的粒子数量.

关键词:粒子滤波;样本贫化;果蝇优化算法;非线性系统;状态估计

中图分类号:TP391.41

文献标志码:A

Improved Particle Filter Based on Fruit Fly Optimization and Its Application in Target Tracking

HAN Kun[†], ZHANG He

(School of Traffic & Transportation Engineering, Central South University, Changsha 410075, China)

Abstract: A particle filter method based on fruit fly optimization algorithm is proposed to alleviate the sample impoverishment caused by resampling. When fruit flies forage, they usually fly from low concentration areas to high concentration areas efficiently and constantly. This optimum process is introduced into the particle filter to drive particles towards the high likelihood areas ceaselessly, and thus improves the overall quality of the particle swarm. Considering that the premature convergence is always associated with the fruit fly optimization algorithm, crossover and mutation operations of genetic algorithms are applied herein adaptively to keep the diversity of samples. Firstly, the particle distribution is improved by cross operation. When the algorithm falls into the local optimum, the Cauchy mutation perturbation is then used to help the fruit fly optimization algorithm jump out of the local optimal point effectively and continue searching for global extremum. The nonlinear simulations and target tracking experiments show that the proposed algorithm improves the estimation accuracy of the nonlinear systems state, and it has better stabil-

* 收稿日期:2017-09-28

基金项目:湖南省自然科学基金资助项目(2016JJ4117), Natural Science Foundation of Hunan Province of China(2016JJ4117); 中南大学中央高校基本科研业务费专项资金资助(2017ZZTS810), Fundamental Research Funds for the Central Universities of Central South University(2017ZZTS810)

作者简介:韩锲(1977-),女,湖北随州人,中南大学副教授,博士

[†] 通讯联系人, E-mail: hkun@csu.edu.cn

ity and reduces the number of particles required for state estimation at the same time.

Key words: particle filter; sample impoverishment; fruit fly optimization algorithm; nonlinear systems; state estimation

粒子滤波是一种用于求解后验概率的方法,它通过非参数化蒙特卡洛模拟方法实现递推贝叶斯滤波^[1].由于粒子滤波摆脱了传统解决非线性滤波问题时,随机量必须满足高斯分布的制约条件^[2],使其能够在非线性、非高斯系统下,相较于基于卡尔曼滤波框架的滤波算法,展现出明显的优越性.目前粒子滤波在无线定位、金融与经济学、参数估计、目标跟踪^[3-6]等领域,都具有非常广阔的应用前景.然而,传统的粒子滤波还存在着一些不足,需要加以改善,例如序列重要性采样会带来粒子退化问题,即经过若干次循环递推后,粒子集中绝大多数粒子权值变得很小,甚至接近于零,只有少数粒子具有非零权值,造成粒子集无法表达实际的后验概率分布^[7].虽然通过重采样^[8]复制权重较大粒子、删除低权值粒子能够在一定程度上抑制粒子退化问题,却在同时也破坏了粒子多样性,导致粒子出现贫化现象^[9].

近年来,为了解决粒子滤波算法存在的这些问题,国内外学者提出了很多改进方法,主要是以优化重采样的角度来改进滤波精度.例如, Li 等^[10]提出确定性重采样,利用粒子空间信息和状态值进行重采样,避免盲目丢弃低权重的粒子,从而保证了粒子的多样性,但是对于权值较大的粒子,直接进行了权值平均,并没有有效利用它们所具有的信息.程水英等^[11]提出在重采样前进行权值排序列、裂变繁殖、权值归一的预处理,以平滑权值差异使得更多的样本能够进行重采样,这样虽然能延迟贫化时间,但无法从根本上避免贫化.张光等^[12]采用正则化方法,引入核密度函数和核带宽系数以连续形式计算状态后验概率,有效缓解粒子退化问题,但无法保证样本粒子都能近似表示后验概率,因而对非线性系统参数估计不能达到最优.

将群体智能优化算法与粒子滤波结合是目前粒子滤波发展的一个较新的思路^[13].采用智能优化算法来避免粒子贫化问题,主要是将标准粒子滤波中的每个粒子看作生物群体的每个个体,通过模拟生物集群的运动规律来调整粒子的分布,使粒子群体向高似然区域移动,更加接近真实后验分布,由于其过程并未舍弃权重低的粒子,因而能够从根本上避免粒子贫化现象^[14].目前,已经有多种基于群智能

优化算法的粒子滤波被提出,例如,田梦楚等^[14]提出基于萤火虫算法的粒子滤波,引入了萤火虫群体的优胜劣汰机制以及萤火虫个体的吸引和移动的行为,使粒子向高似然区域移动,从而提高估计精度.汪荣贵等^[15]将自适应遗传算法应用到粒子滤波相,依据粒子的适应度值自适应确定对粒子进行遗传操作的概率,然后对选出的粒子实施交叉、变异操作来移动粒子,增加粒子多样性. Tian 等^[16]将人工鱼群算法和无迹粒子滤波结合,利用人工鱼群算法优化采样过程,克服样本贫化问题.

果蝇优化算法^[17]由台湾学者潘文超博士于 2011 年提出,是通过模拟自然界果蝇觅食行为而推演出来的群智能优化算法,也是迄今为止所需调整参数最少、进化方程最为简单的群体智能优化算法之一^[18].本文结合果蝇优化算法迭代寻优机制以及粒子滤波的特点,对果蝇优化算法寻优机制加以改进,引入自适应交叉和变异操作,保障粒子多样性,并将结合交叉、变异的果蝇优化算法到融入粒子滤波当中,提出基于改进的果蝇优化思想的粒子滤波算法.通过模型仿真以及目标跟踪实验,表明本文所提算法在保证粒子多样性的同时也能够很好的提高状态估计精度与稳定性.

1 粒子滤波算法

粒子滤波是一种基于蒙特卡罗以及递推贝叶斯估计的统计滤波方法^[7].所谓递推贝叶斯估计,就是根据已有的先验概率分布递推出系统状态的后验概率分布,不断调整粒子的权值和位置来修正先验分布的过程.粒子滤波关键思想在于从先验概率分布中抽取一组随机带权值的样本 $\{x_{k-1}^i, i = 1, \dots, N\}$, 近似表征后验概率密度函数,将推导过程中的积分运算转变为有限样本点的求和运算.后验密度函数近似公式为:

$$p(x_k | z_{1:k}) \approx \sum_{i=1}^{N_k} \omega_k^i \delta(x_k - x_k^i) \quad (1)$$

式中: $\delta(x)$ 是狄拉克函数, x_k 为 k 时刻的状态值, $z_{1:k}$ 为 1 到 k 时刻的观测值.从后验概率密度函数中采样往往是很困难的,一般选择从一个替代的

易采样的重要密度函数 $q(x_k^i | x_{k-1}^i, z_k)$ 中采集样本, 则递推过程粒子权值更新公式为:

$$w_k^i = w_{k-1}^i \frac{p(z_k | x_k^i) p(x_k^i | x_{k-1}^i)}{q(x_k^i | x_{k-1}^i, z_k)} \quad (2)$$

将权值归一化后, 状态输出为:

$$x_k = \sum_{i=1}^N w_k^i x_k^i \quad (3)$$

2 果蝇优化算法(FOA)

果蝇优化算法是一种源于果蝇觅食行为推演出寻求全局优化的群体智能算法^[17]. FOA的仿生原理是将种群数量为 N 的果蝇个体映射为搜索空间中的 N 个可行解, 整个迭代寻优以及搜索过程模拟成果蝇群体觅食过程. 果蝇本身在感官知觉上优于其它物种, 尤其是在嗅觉与视觉上, 果蝇的嗅觉器官能够较好地搜集漂浮在空气中的各种气味, 根据果蝇个体感知到的食物味道浓度的大小来衡量每个果蝇个体所处位置的优劣, 味道浓度越大, 则表明该果蝇位置距离食物越近. 果蝇觅食的过程就是果蝇不断从浓度低的地方飞向浓度高的地方, 飞近食物位置后亦可使用敏锐的视觉发现食物与同伴聚集的位置, 且往该方向飞去, 直到找到食物.

依据果蝇搜索食物的特征行为, 果蝇优化算法基本流程主要由以下 3 个步骤构成:

1) 种群初始化: 确定果蝇群体数量 N 、最大迭代次数 m 以及初始化果蝇个体觅食位置 X_0, Y_0 .

2) 觅食活动: 果蝇群体从初始位置出发, 利用嗅觉搜寻食物, 每个个体的搜索方向与距离皆为设定范围的随机数. 根据当前的位置 X_i, Y_i , 计算每个果蝇适应度值 f_i (味道浓度值), 找出其中适应度值最大(或最小)的个体, 然后群体利用视觉向该个体位置飞去.

3) 种群位置更新: 每一次迭代过程即每一次从当前最佳适应值位置飞出到下一最佳位置的过程, 若下一次最佳适应度值大于前一次最佳适应度值, 则群体更新最佳适应度值点为新的觅食位置 X_0, Y_0 , 所有个体向该位置飞去, 然后再飞出继续搜索. 如此反复, 直到达到最大迭代次数或设定精度为止.

3 果蝇优化粒子滤波(FOA-PF)

3.1 整体改进原理

传统粒子滤波算法的重采样过程通过复制大权

重粒子、删除小权值粒子来解决粒子匮乏现象, 但经过多次迭代后, 又会带来粒子贫化问题.

针对上述问题, 本文将果蝇优化算法融入到粒子滤波中改善采样过程. 算法具体思路如下: 在标准粒子滤波过程中, 当经过重要性采样得到 N 个粒子后, 根据个体的位置以及设定的适应度函数计算每个粒子的适应度值, 如果粒子集分布在真实状态附近, 那么, 粒子群中每个粒子的适应度值都很高; 反之, 如果粒子群中的适应度最优值很低, 则说明粒子集没有分布在真实状态附近, 此时, 利用 FOA 算法对粒子分布进行优化, 粒子不断向适应度值高的地方飞去, 使得粒子不断向真实状态靠近, 从而提高粒子群整体样本的质量. 当粒子集的最优值达到设定的阈值 ϵ 时, 则说明粒子集已经分布在真实状态附近, 此时即可停止优化.

通过上述优化过程, 驱使粒子集合不断向高似然区域靠近, 从而解决粒子贫乏问题. 然而, 果蝇优化算法也存在着一些不可避免的问题, 即在整个寻优过程中果蝇只向当前味道浓度最高的个体飞去, 所有粒子分布随机在最优值附近, 无法保证粒子多样性. 而且如果该个体不是全局最优值, 极易使算法陷入局部最优, 从而降低收敛速度和收敛精度, 带来早熟收敛的问题. 因而直接将果蝇优化算法融入到粒子滤波中并不一定会得到理想的效果, 需要对果蝇优化过程进行适当的改进, 采用一种机制, 让算法在出现早熟收敛时, 能够跳出局部最优, 并进入解空间的其它区域继续进行搜索, 直到找出全局最优解^[19]. 本文采取在果蝇优化算法中引入自适应交叉算子、自适应变异机制, 首先对粒子群体进行自适应交叉操作, 增加粒子多样性, 可以在一定程度减小陷入局部最优几率. 然后判断算法是否陷入早熟收敛, 若是, 则复制一些当前最优粒子, 对这些粒子进行柯西变异操作, 可以促使算法跳出局部最优, 继续搜索全局极值.

3.2 适应度函数设计

标准果蝇优化算法是以果蝇个体与坐标原点的距离的倒数作为果蝇个体味道浓度判定值, 而为了与粒子滤波相结合, 并保证滤波的精度, 本文将最新的量测值引入采样过程, 定义适应度函数为:

$$f(i) = \frac{1}{\sqrt{2\pi\eta_k}} \exp\left(-\frac{(z_k - z_{vic}(i))^2}{2\eta_k}\right) \quad (4)$$

式中: η_k 为测量噪声方差; z_k 为最新观测值; $z_{vic}(i)$ 为观测预测值.

3.3 自适应交叉因子

交叉因子是遗传算法中很重要的一类算子,影响着算法的收敛性. 自适应调节或优化遗传操作概率等控制变量是提高遗传算法性能非常重要的手段^[15],其核心思想是通过个体的适应度函数值来确定交叉概率. 在果蝇优化算法中引入自适应交叉因子,对种群进行交叉操作,产生的新种群能够像自然进化产生的后代一样,会比上一代更加能适应环境的变化,有利于找出全局最优值.

设果蝇优化算法第 t 代种群适应度大小依次为 $f_i (i = 1, \dots, N)$, 则交叉概率按如下公式计算得出^[15]:

$$P = \begin{cases} p_1 - \frac{(p_1 - p_2)}{1 + \exp(-2A \frac{f_b - f_{\text{mean}}}{f_{\text{max}} - f_{\text{mean}}})}, & f_b \geq f_{\text{mean}} \\ p_2, & f_b < f_{\text{mean}} \end{cases} \quad (5)$$

上式中: p_1, p_2 表示交叉概率的变化范围; f_b 表示两个交叉个体中较大的适应度值; f_{max} 是种群中最大适应度值; f_{mean} 为种群适应度平均值; 常数 A 在适应度值高于平均水平时可以对 P 的变化趋势进行调节,使得 P 的变化比较缓慢. 根据上式,可以使得适应度值接近最优的那部分个体也可以保持一定的交叉概率,因为这部分个体经过交叉运算以后,优势基因遗传到子代的可能性更大一些,这一改进可增加新产生的子代个体中优秀个体的数量.

标准的 FOA 算法中,每次得到新一代种群后,引入遗传算法中的交叉因子对该种群进行交叉操作. 具体作法为:将果蝇群体随机两两配对,任意选取一对个体作为父代,根据式(5)确定交叉概率,然后进行和遗传算法相同的交叉操作,产生和父代同样数目的子代进入下一步果蝇优化过程. 通过交叉操作可以增强果蝇群体的多样性,从而增强跳出局部最优的能力. 交叉操作公式如下:

$$\begin{cases} x_k^*(m) = \alpha x_k(m) + (1 - \alpha)x_k(n) \\ x_k^*(n) = \alpha x_k(n) + (1 - \alpha)x_k(m) \end{cases} \quad (6)$$

3.4 自适应变异机制

果蝇群体经过自适应交叉操作后,虽然在一定程度上提高跳出局部最优的能力,但是并不能完全避免早熟情况的发生. 此时根据果蝇群体适应度方差 σ^2 以及当前适应度最优值大小来判定算法是否陷入局部最优^[20]. 其中 σ^2 反映了果蝇群体中所有个体适应度的整体变化情况,其值越小,说明群体的聚集程度越高,种群多样性越低,而最优适应度值越

大,表示越接近真实位置,反之,表示群体与真实位置偏离较大. 若 $\sigma^2 \leq \delta$ (设定方差阈值)且当前适应度最优值的大小并未达到理论最优值或目标精度时,则可判定算法陷入局部最优^[20]. 此时,将迭代的最优果蝇复制 K 个,然后对复制的果蝇按一定的概率 P 进行柯西扰动变异操作,然后对变异后的最优个体进行二次寻优. 本文设计柯西变异公式如下:

$$x_a(j) = x_0(1 + 0.5 \tan(\text{rand}(1, 1) - 0.5)\pi), \quad j = 1, 2, \dots, K \quad (7)$$

式中: $x_a(j)$ 是变异个体的位置; x_0 是当前最优味道浓度值的粒子所在位置. 柯西变异能够提高算法的全局搜索能力,如果变异后的 K 个个体最好适应度值优于当前种群最好适应度值,则以变异后最优的位置作为当前代果蝇群体最优位置进入下一步迭代中. 通过自适应变异操作,在增加种群多样性的同时,能够快速搜索到全局最优点.

3.5 算法步骤

步骤 1. 初始化参数:粒子数 N , 果蝇优化最大迭代数 m , 适应度方差阈值 σ^2 , 若陷入局部最优,复制最优果蝇个体数 K , 变异概率 P .

步骤 2. 初始时刻,采样 N 个粒子 $x_i (i = 1, \dots, N)$ 作为算法初始粒子.

步骤 3. 重要性采样, x_k^i 服从重要性密度函数: $x_k^i \sim q(x_k^i | x_{k-1}^i, z_k) = p(x_k^i | x_{k-1}^i)$ (8)

步骤 4. 对种群分布进行改进的果蝇优化.

1) 根据公式(4)计算每个粒子的适应度值,找出其中适应度最大 f_{max} 的粒子,记该点坐标为 x_0 , 所有粒子向该点飞去,根据公式(9)赋予每个粒子随机方向与距离.

$$x_k^i = x_0 + (2\text{rand} - 1) \times h \quad (9)$$

式中 $(2\text{rand} - 1)$ 为 -1 至 1 之间的一个随机数,表示果蝇搜索的随机方向, h 为最大搜索半径.

2) 根据适应度函数重新计算当前种群适应度值,然后根据公式(5)、(6)进行自适应交叉操作.

3) 计算交叉后的个体适应度值 f_i 、最优值 f_{max} 、群体平均适应度值 f_{mean} 粒子群体味道浓度方差 σ^2 .

$$f_{\text{mean}} = \frac{1}{N} \sum_{i=1}^N f(i) \quad (10)$$

$$\sigma^2 = \sum_{i=1}^N (f(i) - f_{\text{mean}})^2 \quad (11)$$

4) 若 $\sigma^2 \leq \delta$ (设定的方差阈值)、当前最佳适应度值小于理论最大值或是目标精度,则判断算法陷入局部最优. 若分布于 0 与 1 之间的随机数小于变

异概率 P 时,则复制 K 个当前最优个体,然后按照公式(7)对复制的个体进行柯西变异扰动操作,更新它们的位置.

5)计算变异后的个体的适应度值,记录并保留适应度最大的粒子及其位置.

6)若变异后的最大适应度 $f_{\max} \geq f_{\text{best}}$,则更新最佳适应度值为:

$$f_{\text{best}} = f_{\max} \quad (12)$$

7)进入迭代寻优,重复执行果蝇优化步骤 1~6,直到迭代步数达到最大迭代次数 m ,或者符合设定的精度阈值时,停止优化.

步骤 5. 保留果蝇优化最后一次迭代的粒子群体,更新权重值.

$$\omega_k^i = \omega_{k-1}^i p(z_k | x_k^i) \quad (13)$$

步骤 6. 归一化权值

$$\omega_k^i = \frac{\omega_k^i}{\sum_{i=1}^N \omega_k^i} \quad (14)$$

步骤 7. 状态输出

$$x_k^* = \sum_{i=1}^N \omega_k^i X_k^i \quad (15)$$

步骤 8. 时刻 $t = t + 1$,转到步骤 3.

上述步骤充分利用了整个粒子集中的有效信息,驱使粒子向高似然区域移动.利用自适应交叉操作,可以使得果蝇优化过程粒子分布更为合理,通过变异操作,增加粒子集多样性,促进粒子跳出早熟并继续搜索.由于果蝇优化算法的收敛效率相对较高,如果在每次迭代过程中,FOA-PF 都能够达到最终收敛,那么所有粒子都将集中在收敛位置,这样反而会降低粒子的多样性.因而本文通过设置阈值以及最大迭代次数,提前结束粒子寻优过程,然后取经交叉操作后的粒子集作为粒子滤波每一时刻的最终粒子分布,这样既提高算法计算效率,使得粒子群体有向最优区域靠拢的趋势,又可以防止最终收敛,从而保证 FOA-PF 在精度和效率方面的优势.

4 模型仿真实验

实验硬件为英特尔 i7 处理器、8G 内存,软件环境为 Matlab2014b.在目标跟踪算法中,均值漂移、相关滤波等,都是直接作用于实际视频当中,无法跟踪一个设定的数学模型.对于数学模型的跟踪,当前主要采用的还是贝叶斯滤波的方法,由于粒子滤波算法相较于早期的卡尔曼或者卡尔曼改进形式如扩

展卡尔曼等,在非线性系统预测精度上具有较大优势,因而为了验证本文算法的有效性,只对比了同类算法,进行了标准粒子滤波(PF)、遗传算法优化粒子滤波(GA-PF)、基于果蝇优化思想的粒子滤波(FOA-PF)分别在粒子总数为 20、50 和 100 时的滤波实验.本文采用单变量非静态增长模型作为仿真对象,其具有高度的非线性,而且似然函数呈双峰状,这使得传统的滤波方法很难处理.系统模型如下式所示.

过程模型:

$$x_k = 0.5x_{k-1} + \frac{20x_{k-1}}{1+x_{k-1}^2} + 8\cos[1.2(k-1)] + w_k \quad (16)$$

量测模型:

$$z_k = 0.2x_k^3 + \frac{x_k^2}{13} + v_k \quad (17)$$

式中: w_k 和 v_k 皆为零均值高斯噪声,设系统噪声 w_k 方差 $Q = 1$,测量噪声 v_k 方差 $R = 1$,迭代次数 T 为 50,果蝇优化粒子滤波中寻优最大迭代数 m 为 20,取迭代步长 $h = 4$,适应度方差阈值 $\delta = 0.5$,当陷入早熟时,复制最优个体粒子数分别为 10、20、50.遗传算法优化粒子滤波中交叉概率为 0.7,变异概率为 0.05.

4.1 精度和稳定性测试

取均方根误差作为算法估计精度的判定值,均方根误差公式为:

$$\text{RMSE} = \left[\frac{1}{T} \sum_{i=1}^T (x_k - \tilde{x}_k)^2 \right]^{1/2} \quad (18)$$

以均方值误差的方差作为算法的稳定性判定依据,方差越低,表明算法稳定性越好.选取 PF、GA-PF、FOA-PF 三种算法进行仿真实验,状态估计以及误差结果如图 1 至图 6 所示.

其中:图 1 和图 2 所示为粒子数 $N = 20$ 的状态估计与误差绝对值仿真结果;

图 3 和图 4 所示为粒子数 $N = 50$ 的状态估计与误差绝对值仿真结果;

图 5 和图 6 所示为粒子数 $N = 100$ 的状态估计与误差绝对值仿真结果.

从图 1~6 可以看出,本文所提基于改进的果蝇优化算法的粒子滤波(FOA-PF),相较于标准 PF 以及 GA-PF,状态预测曲线与实际状态相似程度最高,其估计值更接近真实值,这是因为 FOA-PF 在 PF 的基础上,通过对重要性采样后的粒子进行引入交叉、变异操作的果蝇迭代寻优,使粒子向高似然区

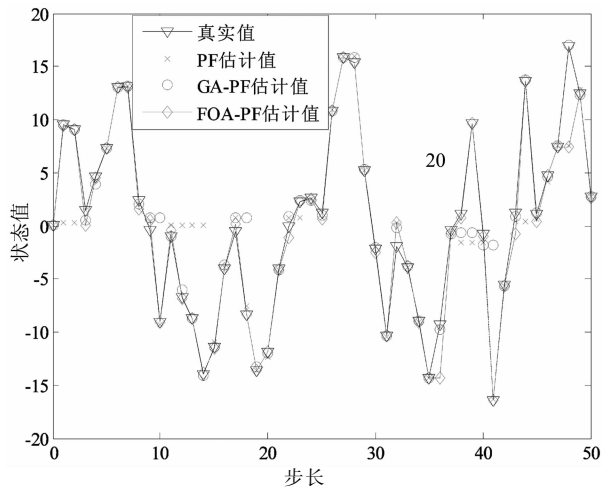


图 1 状态估计(N=20)
Fig. 1 State estimate(N=20)

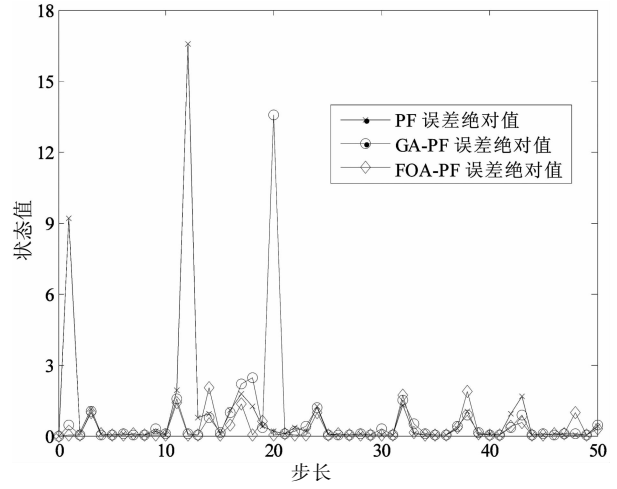


图 4 滤波误差绝对值(N=50)
Fig. 4 Absolute value of filter error(N=50)

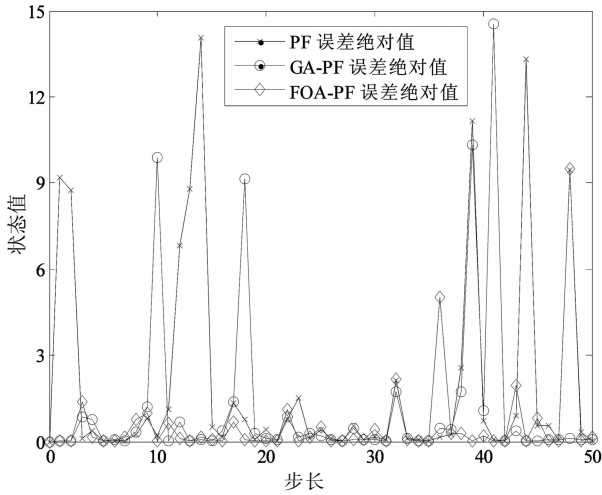


图 2 滤波误差绝对值(N=20)
Fig. 2 Absolute value of filter error(N=20)

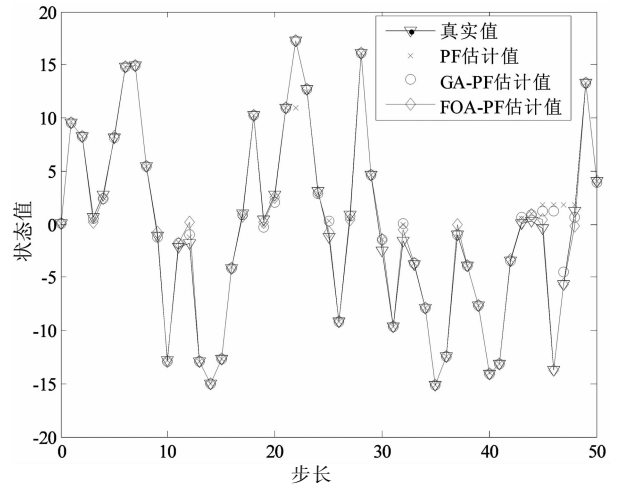


图 5 状态估计(N=100)
Fig. 5 State estimate(N=100)

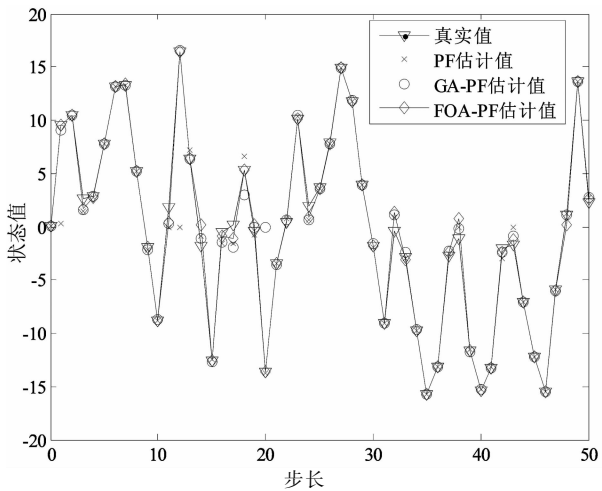


图 3 状态估计(N=50)
Fig. 3 State estimate(N=50)

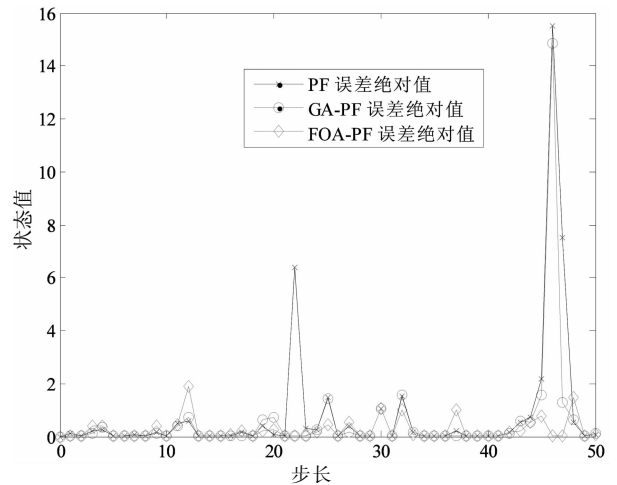


图 6 滤波误差绝对值(N=100)
Fig. 6 Absolute value of filter error(N=100)

域运动的同时,保证了样本多样性,从而提高粒子分布的合理性.表1、表2中数据为每种算法独立运行50次后的均方根误差平均值以及方差,从中可以看出,3种算法随着粒子数的增加,均方根误差以及方差大体上皆呈现减小的趋势,这与粒子滤波的粒子数越多则估计精度越高的理论是相符的.而在三种算法中,FOA-PF的均方根误差以及方差最低,且是唯一一个在粒子数为100时两者皆小于1的,当粒子数为50时,其误差值也比粒子数为100的PF算法低,与GA-PF也只有不到0.2的差距,说明FOA-PF能够用较少的粒子达到所需的精度.而均方根误差方差体现算法预测的稳定性,FOA-PF也是最低的,当粒子数越多时更为明显.综上分析,FOA-PF具有更好的估计精度以及稳定性.

表1 估计精度对比

Tab.1 Comparison of estimation accuracy

参数	RMSE 均值		
	PF	GA-PF	FOA-PF
N=20	3.993 2	3.360 3	2.646 1
N=50	2.599 1	1.735 5	1.414 4
N=100	1.542 6	1.246 1	0.729 5

表2 估计稳定性对比

Tab.2 Comparison of estimation constancy

参数	RMSE 方差		
	PF	GA-PF	FOA-PF
N=20	1.862 7	1.676 5	1.389 7
N=50	1.599 8	1.499 5	1.291 3
N=100	1.250 9	1.189 4	0.454 4

4.2 粒子多样性测试

为测试FOA-PF滤波时的粒子多样性,设置粒子数为100,取PF和FOA-PF滤波器在迭代过程第20和45步时的粒子分布情况,如图7~8所示,可以看出,FOA-PF与标准PF相比,具有更宽的粒子分布,除了在高似然区域存在较多粒子,周边区域也分布着部分粒子,说明FOA-PF在预测精度优于PF的同时,粒子多样性并没有降低,这是因为对果蝇寻优过程进行了自适应的交叉与变异操作,保证了粒子多样性不下降,而且本文并没有进行重采样操作,也能够有效地避免粒子贫化现象的出现.

5 目标跟踪实验

目标跟踪在计算机视觉领域中有着广泛的应用前景和商业价值,包括智能监控、人机交互、视频检

索等.快速移动的目标往往具有非线性、非高斯的特点,跟踪要求有较高的实时性、鲁棒性.近年来,基于相关滤波^[2,1]的判别式跟踪方法以其优越的跟踪速度、出色的跟踪性能及高效的鲁棒性得到了广泛关注和应用.为检验本文所提算法在目标跟踪应用中的实际效果,将本文所提方法与相关滤波算法分别应用到视频目标跟踪当中,并对结果进行分析.

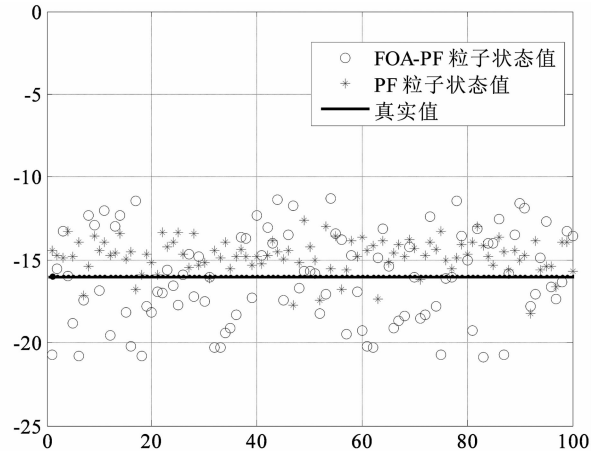


图7 k=20时粒子分布图

Fig.7 Particle distribution when k=20

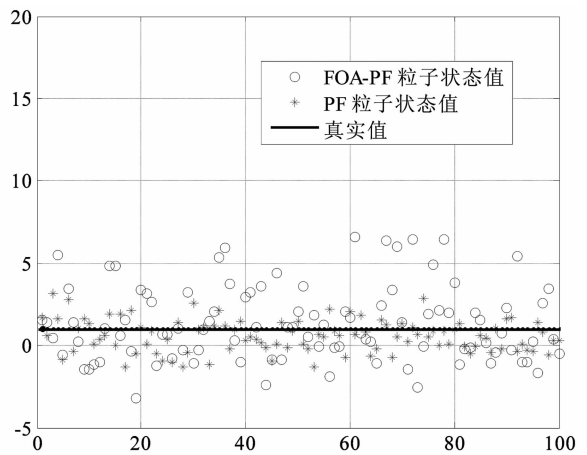


图8 k=45时粒子分布图

Fig.8 Particle distribution when k=45

视频目标的跟踪需要选择合适的特征来描述该目标.其中,颜色特征是图像各类特征中最显著且最稳定的视觉特征,本文采用加权颜色直方图来描述颜色信息,目标区域的加权颜色直方图分布为:

$$p_y^{(a)} = f_{color} \sum_{i=1}^N k \left(\frac{\|y - x_i\|}{a} \right) \delta[h(x_i) - u] \tag{19}$$

式中: f_{color} 为归一化常数; N 为选择目标区域所包含的像素数; a 为目标区域的大小; $\delta(x)$ 为狄拉克函数.设目标模型颜色直方图为 p , 粒子区域颜色

直方图为 q , 由 Bhattacharyya 系数来衡量两者之间相似度 $\rho(p, q)$ 、距离 d , 表示如下:

$$\rho(p, q) = \sum_{i=1}^S \sqrt{p(i)q(i)} \quad (20)$$

$$d(p, q) = \sqrt{1 - \rho(p, q)} \quad (21)$$

观测概率密度函数表示如下:

$$p(y | x_k^i) = \frac{1}{\sqrt{2\pi\theta_c}} \exp\left(-\frac{d^2}{2\pi\theta_c^2}\right) \quad (22)$$

则权值更新公式为:

$$\omega_k^i = \omega_{k-1}^i p(y | x_k^i) \quad (23)$$

粒子区域和目标区域之间颜色直方图相似度越高, 则该区域为真实目标的可能性越大。

实验素材选取 Visual Tracker Benchmark 网站上 David3 和 Girl 两个视频, 其中 David3 中目标沿着直线运动, 其中有两处被暂时遮挡; Girl 视频取中间 220 帧作为实验素材, 跟踪目标为女生的脸部, 整个视频跟踪过程中, 女生脸部做随机运动, 从第 144 帧开始出现遮挡情况, 定义目标在跟踪框内面积少于一半时为跟踪偏离, 完全不在框内为跟踪丢失。两种情景下相关滤波与 FOA-PF 两种算法跟踪效果分别如图 9~图 12 所示。

在两种跟踪场景下, 相关滤波的跟踪速度都占有绝对的优势, 粒子滤波由于需要不断的迭代使得运行时间要高于相关滤波算法。而在跟踪精度方面, 第一个实验中两者没有明显差异, 而在第二个视频跟踪过程中, FOA-PF 全程没有跟丢, 偏离的情况也几乎没有发生, 而相关滤波在出现遮挡情况后, 跟踪了错误的目标。实验表明 FOA-PF 在遮挡与快速移动的情况下, 有着更好的鲁棒性。

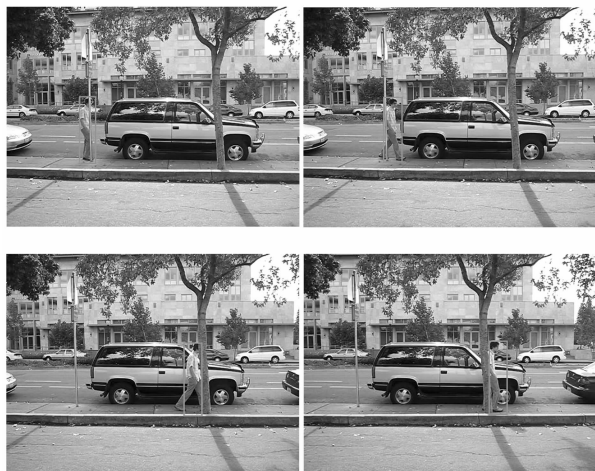


图 9 相关滤波第 23、28、80、86 帧跟踪实验结果
Fig. 9 Correlation filter tracking results of the 23 th, 28 th, 80 th and 86 th frame

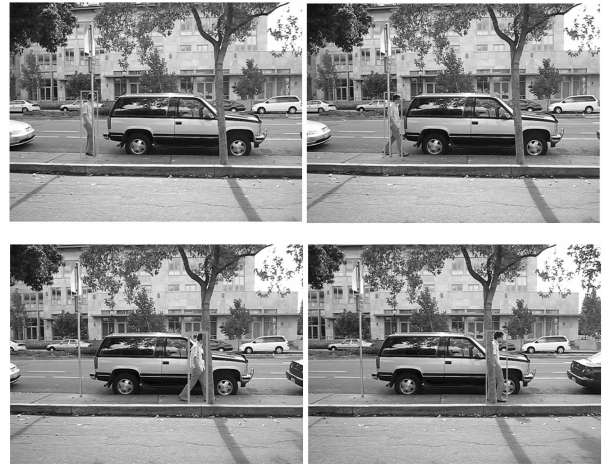


图 10 FOA-PF 算法第 23、28、80、86 帧跟踪实验结果
Fig. 10 FOA-PF tracking results of the 23 th, 28 th, 80 th and 86 th frame



图 11 相关滤波第 50、125、149、192 帧跟踪实验结果
Fig. 11 Correlation filter tracking results of the 50 th, 125 th, 149 th and 192 th frame



图 12 FOA-PF 算法第 50、125、149、192 帧跟踪实验结果
Fig. 12 FOA-PF tracking results of the 50 th, 125 th, 149 th and 192 th frame

6 结 论

1)本文算法 FOA-PF 通过单变量非静态增长模型进行了仿真,粒子数为 100 时,实验结果均方根误差均值为 0.729 5,均方根误差方差为 0.454 4,表明本文算法精度、稳定性均高于标准粒子滤波和基于遗传算法的粒子滤波算法,粒子分布图进一步说明本文算法有效保证了粒子的多样性。

2)将 FOA-PF 算法运用到视频目标跟踪实验中,与相关滤波算法跟踪结果进行比较,实验结果表明本文算法对目标快速移动以及被遮挡的情况都具有良好的跟踪效果。

参考文献

- [1] DOUCTE A, DE FREITAS N, GORDON N. Sequential Monte Carlo methods in practice[M]. New York: Springer-Verlag, 2001.
- [2] 王法胜, 鲁明羽, 赵清杰, 等. 粒子滤波算法[J]. 计算机学报, 2014, 37(8):1679-1694.
WANG F S, LU M Y, ZHAO Q J, *et al.* Particle filter algorithm[J]. Journal of Computer Science, 2014, 37(8):1679-1694. (In Chinese)
- [3] GUSTAFSSON F. Particle filter theory and practice with positioning applications[J]. Aerospace & Electronic Systems Magazine IEEE, 2010, 25(7):53-82.
- [4] 李天成, 范红旗, 孙树栋. 粒子滤波理论、方法及其在多目标跟踪中的应用[J]. 自动化学报, 2015, 41(12):1981-2002.
LI T C, FAN H Q, SUN S D. Particle filtering: theory, approach, and application for multitarget tracking[J]. Acta Automatica Sinica, 2015, 41(12): 1981-2002. (In Chinese)
- [5] GAO M, ZHANG H. Sequential Monte Carlo methods for parameter estimation in nonlinear state-space models[J]. Computers & Geosciences, 2012, 44(13):70-77.
- [6] CREAL D. A survey of sequential Monte Carlo methods for economics and finance[J]. Serie Research Memoranda, 2009, 31(3):245-296.
- [7] 朱志宇. 粒子滤波算法及其应用[M]. 北京: 科学出版社, 2010:27-31.
ZHU Z Y. Particle filter algorithm and its application[M]. Beijing: Science Press, 2010:27-31. (In Chinese)
- [8] ARULAMPALAM M S, MASKELL S, GORDON N, *et al.* A tutorial on particle filters for online nonlinear/nongaussian Bayesian tracking[J]. IEEE Transactions on Signal Processing, 2002, 50(2):174-188.
- [9] FOO P H, NG G W. Combining the interacting multiple model method with particle filters for manoeuvring target tracking[J]. IET Radar, Sonar and Navigation, 2011, 5(3):234-255.
- [10] LI T C, SATTAR T P, SUN S D. Deterministic resampling: unbiased sampling to avoid sample impoverishment in particle filters[J]. Signal Processing, 2012, 92(7):1637-1645.
- [11] 程水英, 张剑云. 裂变自举粒子滤波[J]. 电子学报, 2008, 36(3):500-504.
CHENG S Y, ZHANG J Y. Fission bootstrap particle filter[J]. Journal of Electronics, 2008, 36(3):500-504. (In Chinese)
- [12] 张光, 张英堂, 任国全, 等. 基于正则化粒子滤波的磁梯度张量跟踪方法[J]. 探测与控制学报, 2014(2):0050-0053.
ZHANG G, ZHANG Y T, RENG G Q, *et al.* Tracking method of magnetic gradient tensor based on RPF[J]. Journal of Detection & Control, 2014(2):0050-0053. (In Chinese)
- [13] YU Y, ZHENG X. Particle filter with ant colony optimization for frequency offset estimation in OFDM systems with unknown noise distribution[J]. Signal Processing, 2011, 91(5):1339-1342.
- [14] 田梦楚, 薄煜明, 陈志敏, 等. 萤火虫算法智能优化粒子滤波[J]. 自动化学报, 2016, 42(1):89-97.
TIAN M C, BO Y M, CHEN Z M, *et al.* Firefly algorithm intelligence optimized particle filter[J]. Acta Automatica Sinica, 2016, 42(1):89-97. (In Chinese)
- [15] 汪荣贵, 李孟敏, 吴昊, 等. 一种新型的基于自适应遗传算法的粒子滤波算法[J]. 中国科学技术大学学报, 2011, 41(2):134-141.
WANG R G, LI M M, WU H, *et al.* A new particle filter algorithm based on the adaptive genetic algorithm[J]. Journal of University of Science and Technology of China, 2011, 41(2):134-141. (In Chinese)
- [16] TIAN Y, LU C, WANG Z, *et al.* Artificial fish swarm algorithm-based particle filter for li-ion battery life prediction[J]. Mathematical Problems in Engineering, 2014(3):1-10.
- [17] PAN W T. A new fruit fly optimization algorithm; taking the financial distress model as an example[J]. Knowledge-Based Systems, 2012, 26(2):69-74.
- [18] 韩虎. 果蝇优化算法的分析[J]. 计算机系统应用, 2017, 26(2):9-17.
HAN H. Analysis on fruit fly optimization algorithm[J]. Journal of Computer Applications, 2017, 26(2):9-17. (In Chinese)
- [19] 吕振肃, 侯志荣. 自适应变异的粒子群优化算法[J]. 电子学报, 2004, 32(3):416-420.
LÜ Z S, HOU Z R. Particle swarm optimization with adaptive mutation[J]. Acta Electronica Sinica, 2004, 32(3):416-420. (In Chinese)
- [20] 韩俊英, 刘成忠. 自适应变异的果蝇优化算法[J]. 计算机应用研究, 2013, 30(9):2641-2644.
HAN J Y, LIU C Z. Fruit fly optimization algorithm with adaptive mutation[J]. Application Research of Computers, 2013, 30(9):2641-2644. (In Chinese)
- [21] HENRIQUES J F, RUI C, MARTINS P, *et al.* High-speed tracking with kernelized correlation filters[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2014, 37(3):583-596.