

安徽工程大学本科

# 毕业设计（论文）

专          业： 计算机科学与技术

题          目： 图像空域增强技术的设计与实现

作 者 姓 名： 盛 周 璇

导师及职称： 窦易文(副教授)/蒋胜(助教)

导师所在单位： 计算机与信息学院

2019 年 6 月 1 日



# 安徽工程大学

## 本科毕业设计（论文）任务书

2019 届 计算机与信息 学院

计算机科学与技术 专业

学生姓名： 盛 周 璇

### I 毕业设计（论文）题目

中文：图像空域增强技术的设计与实现

英文： **Design and Implementation of Image Enhancement in Spatial Domain**

### II 原始资料

- [1]Magdalene M, Yannis M, Constantin Z. Honey bees mating optimization algorithm for financial classification problems [J]. Applied Soft Computing, 2010, 10(3):806-812.
- [2]Rebreyend P, Clugery C, Hily E. A heuristic -based bee colony algorithm for the multiprocessor scheduling problem [C]// Proc of International Workshop on Nature Inspired Cooperative Strategies for Optimization. Berlin: Springer- Verlag, 2010, 295-304.
- [3]Birbil S T, Fang S C. An electromagnetism-like mechanism for global optimization [J]. Journal of Global Optimization, 2003, 25(3): 263-282.
- [4]肖永豪, 余卫宇. 基于蜂群算法的图像边缘检测[J]. 计算机应用研究, 2010, 27(7): 2748-2750.
- [5]唐超礼, 群智能算法及其在函数优化中的应用研究[D]. 淮南: 安徽理工大学, 2007.

### III 毕业设计（论文）任务内容

#### 1、课题研究的意义

Tubbs 提出的不完全 Beta 函数是空间域图像增强的一种较好方法，能完全覆盖图像增强问题的典型变换函数，增加了图像增强灰度变换函数选取的灵活性，但其包含的  $\alpha, \beta$  参数的选取过程并不直观，没有统一的方法。目前已提出了借助遗传算法、粒子群算法、鱼群算法自适应实现灰度图像增强的思路，但由于图像维数一般较大，而群智能算法在优化高维函数时普遍具备收敛较慢、精度较差的缺陷，故利用群智能算法增强的图像还有很大的提高空间。细菌觅食算法具有较高的搜索精度，强鲁棒性，机理简单，可以更好地弥补其它优化算法的不足。因此，对细菌觅食算法在图像增强中的应用进行研究，并寻求一种更好的最优化问题建模方案，以提高图像增强技术的效率和性能，具有重要的意义。

#### 2、本课题研究的主要内容

细菌觅食(BFO)算法通过个体间的协作和竞争实现全局搜索，系统初始化为一组随机解，初始种群的构造，趋化操作和迁徙操作的设计对算法的性能有较大的影响。因而需要研究如何构建合适的初始种群。

为使 BFO 算法对于复杂函数也能够快速、准确地收敛至全局最优解，同时有效适用于复杂的实际工程最优化问题，本课题拟从三个方面将经典 BFO 应用到图像增强过程中：

- (1) 设计随机化佳点集的菌群生成方法，结合粒子群优化算法改进趋化方向；
- (2) 研究变次数游动和变概率的迁徙策略；
- (3) 将图像增强转化为固定 2 维最优化问题。

#### 3、提交的成果

- (1) 毕业设计（论文）正文；
- (2) 设计出基于空域的图像增强算法；
- (3) 至少一篇引用的外文文献及其译文；
- (4) 附不少于 10 篇主要参考文献的题录及摘要。

指导教师（签字）

教研室主任（签字）

批 准 日 期 2018 年 12 月 20 日

接 受 任 务 书 日 期 2018 年 12 月 25 日

完 成 日 期 2019 年 6 月 1 日

接受任务书学生（签字）



## 图像空域增强技术的设计与实现

### 摘 要

图像空域增强因其作用于像素的直接性、所得结果无需变换的实时性等特点一直是图像处理领域的研究热点。本文研究了传统的图像增强方法，针对空域增强方法缺乏自适应性的缺点，将群智能优化算法中的细菌觅食算法应用于灰度图像增强问题，并研究分析了经典细菌觅食算法的原理优点及缺陷，对其存在的不足之处进行了改进，取得了较好的增强效果。

本课题研究取得的主要成果如下：改进了菌群初始化方法，使用随机化佳点集的方式构造菌群，布点均匀又不失多样性；结合粒子群算法的思想，改进了细菌趋化操作中趋化方向选择策略，并设计了自适应游动次数策略，提高了算法寻优精度和收敛速度；采用自适应迁徙概率机制抑制菌群早熟，避免进化停滞现象，增强了算法寻优性能；用改进的细菌觅食算法优化不完全  $\beta$  函数的两个参数，以自适应拟合四种灰度变换曲线，并设计了合理的适应度目标函数，有效增强了图像细节。

改进后的细菌觅食算法相比经典细菌觅食算法有了很大的性能改善，后续可以继续探究其在其他工程优化问题上的应用，如人工神经网络的训练、图像分割、彩色图像增强、支持向量机等。

**关键词：**群智能；细菌觅食优化算法；图像空域增强；不完全  $\beta$  函数；非线性灰度变换

## Design and Implementation of Image Enhancement in Spatial Domain

### Abstract

Image spatial enhancement has become a research hotspot in this field due to its directness of acting on pixels and its real-time results without transformation. This paper studies the traditional image enhancement method, in view of the spatial domain enhancement method is lack of adaptability shortcomings, bacterial foraging algorithm of swarm intelligence optimization algorithms applied in improvement of gray level transformation, and analysis the advantages and defects, the principle of classical BFO algorithm improves the deficiency of its existence, has obtained the good effect.

The main results obtained in this study are as follows: improved the method of initialization of the flora, using the method of randomized optimal point set to construct the flora, with uniform distribution and diversity; Combined with the idea of particle swarm optimization (pso), the strategy of selecting chemotactic direction in bacterial chemotactic operation is improved, and an adaptive swimming number strategy is designed to improve the optimization accuracy and convergence speed of the algorithm. The adaptive migration probability mechanism was used to inhibit the early maturation of the colony, avoid the phenomenon of evolutionary stagnation, and enhance the optimization performance of the algorithm. Two parameters of incomplete beta function were optimized by improved bacterial foraging algorithm to fit four gray transformation curves adaptively, and a reasonable fitness objective function was designed to enhance the image details effectively.

The improved bacterial foraging algorithm has a great improvement in performance compared with the classical bacterial foraging algorithm, and its application in other engineering optimization problems can be further explored, such as artificial neural network training, image segmentation, color image enhancement, support vector machine and so on.

**Key words:** Swarm Intelligence; Optimization Algorithm of Bacterial Foraging; Image Spatial Enhancement; Incomplete Beta Function; Nonlinear Grayscale Transformation

## 目 录

引 言 .....	- 1 -
第 1 章 绪论 .....	- 2 -
1.1 研究背景 .....	- 2 -
1.2 研究现状 .....	- 3 -
1.3 本文工作 .....	- 4 -
1.4 内容安排 .....	- 4 -
第 2 章 图像增强方法概述 .....	- 6 -
2.1 图像的像素点运算 .....	- 6 -
2.2 空间域图像增强 .....	- 13 -
2.3 频率域增强 .....	- 23 -
2.4 模糊增强 .....	- 23 -
2.5 本章小结 .....	- 24 -
第 3 章 经典细菌觅食优化算法 .....	- 25 -
3.1 群智能优化算法 .....	- 25 -
3.2 细菌觅食算法原理 .....	- 25 -
3.3 细菌觅食算法流程 .....	- 30 -
3.4 细菌觅食算法的优缺点 .....	- 31 -
3.5 本章小结 .....	- 32 -
第 4 章 细菌觅食算法改进研究 .....	- 33 -
4.1 菌群初始化改进 .....	- 33 -
4.2 趋化行为改进 .....	- 34 -
4.3 引入聚集行为 .....	- 37 -
4.4 迁徙行为改进 .....	- 37 -
4.5 算法设计与实现 .....	- 39 -
4.6 本章小结 .....	- 40 -
第 5 章 改进算法应用于灰度图像增强问题 .....	- 42 -
5.1 灰度图像增强问题 .....	- 42 -
5.2 改进算法在灰度图像增强问题上的应用 .....	- 43 -
5.3 仿真结果与分析 .....	- 45 -
5.4 本章小结 .....	- 48 -
结论与展望 .....	- 49 -
致 谢 .....	- 50 -
参考文献 .....	- 51 -



附录 A .....	- 53 -
附录 B .....	- 66 -
附录 C .....	- 72 -

## 插图清单

图 2-1 图像直方图 .....	- 7 -
图 2-2 归一化直方图 .....	- 7 -
图 2-3 灰度线性变换 .....	- 8 -
图 2-4 对数变换 .....	- 9 -
图 2-5 傅里叶频谱图像 .....	- 10 -
图 2-6 伽玛变换 .....	- 11 -
图 2-7 不同亮度和对比度图像直方图均衡化后效果 .....	- 13 -
图 2-8 空域滤波效果图 .....	- 14 -
图 2-9 平均平滑效果 .....	- 15 -
图 2-10 高斯平滑效果 .....	- 16 -
图 2-11 中值滤波效果 .....	- 17 -
图 2-12 Robert 算子锐化效果 .....	- 19 -
图 2-13 Sobel 算子锐化效果 .....	- 19 -
图 2-14 拉普拉斯锐化效果 .....	- 21 -
图 2-15 高斯-拉普拉斯函数的三维形 .....	- 22 -
图 2-16 高斯-拉普拉斯变换效果 .....	- 22 -
图 3-1 细菌觅食行为 .....	- 26 -
图 3-2 趋化操作流程 .....	- 27 -
图 3-3 繁殖操作流程 .....	- 29 -
图 3-4 迁徙操作流程 .....	- 30 -
图 3-5 细菌觅食算法总流程图 .....	- 31 -
图 4-1 粒子群算法中粒子受合力牵引示意图 .....	- 35 -
图 4-2 改进后的迁徙操作流程 .....	- 39 -
图 4-3 改进算法总流程图 .....	- 40 -
图 5-1 四种灰度变换曲线 .....	- 42 -
图 5-2 改进 BFO 的图像增强算法流程图 .....	- 44 -
图 5-3 细菌觅食过程中位置更新示意图 .....	- 46 -
图 5-4 改进 BFO 增强效果对比图（lena） .....	- 47 -
图 5-5 改进 BFO 图像增强效果对比图（cameraman） .....	- 47 -

## 表格清单

表 2-1 三类增强方法的比较 .....	- 24 -
-----------------------	--------

## 引 言

图像增强作为图像预处理的一部分，是计算机视觉中的一个关键技术和经典问题，对后级的模式识别、目标跟踪等诸多领域有着极为重要的应用。目前传统的图像增强技术大致可分为空域法、频域法和模糊处理三大类。空域法以对图像像素的直接操作为基础，如灰度变换、图像平滑和图像锐化等，因其本身具有作用于像素的直接性、所得结果不需进行任何变换的实时性等特点，一直是该领域的研究热点。

本课题的研究将基于生物启发的群智能优化算法引入到图像空域增强技术中，以增加传统图像增强过程中参数选取的智能性，取得了良好的增强效果。在广泛分析评估了近十几年来提出的群智能算法的优缺点及适用领域的基础上，采用了 Passino 提出的细菌觅食优化算法(Bacteria Foraging Optimization, BFO)，其具有算法结构简单易于实现、算法性能稳定、本质并行性及局部搜索能力强且收敛性好等优点，但也存在全局寻优能力较差、在高维多模态问题上易陷入早熟收敛、“精英逃逸”现象等缺点。故本文在经典 BFO 的基础上结合粒子群算法的优点对其进行了改进研究，提出了一种改进算法——具有协同思想的细菌觅食算法。

将改进后算法应用于图像处理技术，针对图像本身的高维特点，将变高维的灰度图像增强问题转换为固定二维的不完全 beta 函数参数寻优问题，以拟合四种灰度变换曲线，设计了全面兼顾图像质量评价指标的目标函数以衡量算法性能。仿真实验结果表明了本文方法的有效性，其增强后的图像与其它方法相比，细节表现更自然，直方图分布更均匀，明暗区域分配更合理。

对于改进后的细菌觅食优化算法，如何合理选择和调整控制参数是影响算法性能的关键，并在高维函数寻优问题上还存在求解精度的不足。后续工作将继续探寻算法参数设置的规律，以改善算法性能。另外，也可以做将该算法应用于其他工程领域的研究。

## 第1章 绪 论

### 1.1 研究背景

图像增强属于图像预处理操作，其目的是消除噪声，根据特定的需要突出一幅图像中用户感兴趣的特征或显现被模糊了的细节，以改善图像的视觉效果，降低图像信息熵，便于对图像进行其他处理。目前传统的图像增强技术大致可分空域法、频域法和模糊处理三大类。空域法以对图像像素的直接操作为基础，如灰度变换、图像平滑和图像锐化等，因其本身具有作用于像素的直接性、所得结果不需进行任何变换的实时性等特点一直是该领域的研究热点，但也存在应用环境窄、算法普适性差等不足；频域法是对原图像进行某种变换（如傅里叶变换），在变换域中进行处理以达到增强目的，缺点是变换参数的选取需要较多的人为干预，且计算量较大；模糊处理是将空域中的原始图像进行模糊化映射，在特征平面中对其进行处理，再逆映射到空间域中以实现增强。这些传统图像增强方法的自适应性和智能性较差，需要较多的人为介入，大大限制了其应用范围。

近几十年来快速发展的群智能优化算法通过对自然界中生物的进化过程和群体性社会行为进行模拟，并将这些过程和行为映射成抽象的搜索和优化过程，应用到实际工程领域的诸多问题中去，取得了令人惊艳的效果，也一直是计算机科学领域研究的热点。群智能优化算法尤其适合于那些目标函数非连续可微、甚至无法定义明确的目标函数的任务，具有对优化函数解析要求低、适合高维、多态等复杂问题的优点。

学术界多年的研究也衍生出了一众群智能算法，如粒子群算法（PSO）源于对鸟类捕食行为的模拟；蜂群算法（ABC）源于对蜜蜂觅食过程中舞蹈行为的观察；蚁群算法（ACO）借鉴了蚁群寻找最短路径中释放信息素的过程。此外，还有一些非群智能的仿生算法如遗传算法、人工神经网络、人工免疫算法<sup>[9]</sup>等，也与群智能算法有异曲同工的思想。与传统的梯度法、牛顿法等一系列确定性优化方法不同，群智能算法无需梯度信息，是一种强随机性、高鲁棒性、高并行性的概率搜索算法，在实际工程领域应用广泛，具有重要的应用研究意义。

Passino 受人体大肠杆菌觅食行为的启发，提出了细菌觅食优化算法，模拟了大肠杆菌的觅食规律，通过细菌群体间的竞争与协作、繁殖和进化等过程实现优化，是一种简单有效的随机全局优化算法，因其群体并行性和局部搜索能力强等优点受到工程界和学术界的关注<sup>[1]</sup>。但其提出时间较晚，研究的不多，还存在不完善的地方，如在优化一些易陷入早熟的高维多峰函数时，其求解精度和稳定度还有很大改进空间，该算法在理论和工程应用方面仍有很多问题需要更深入的研究。

Tubbs 提出的不完全 beta 函数是解决图像空域增强问题的一种较好的方法，能完全覆盖灰度图像增强的四种典型变换曲线<sup>[2]</sup>，增加了灰度变换函数选取的灵活性和自适应性。但其包含的 $\alpha, \beta$ 参数的选取过程并不直观，传统的方法一般是穷举法或人工介入，目前已提出了利用遗传算法、粒子群算法来自适应拟合灰度图像变换曲线的思路，但由于图像的维数一般较大，群智能在优化高维函数时普遍具有收敛速度慢、精度低的缺陷，因此对细菌觅食算法在图像增强问题中的应用进行研究，寻求一种更好的最优化问题建模方案，以提高图像增强技术的效率和性能，具有重要的意义。

## 1.2 研究现状

图像处理和分析的智能化和自动化一直是图像处理学科研究热点之一,也是一个亟待解决的关键问题;为解决灰度图像增强过程中的自适应性问题,多引入仿生算法来寻找变换函数的最优参数,前人在使用遗传算法、粒子群算法等经典智能算法解决图像增强问题也取得了较多的研究成果;但一般的智能优化算法由于算法较为单一,寻优效果不尽完善,会出现局部搜索不精确、易发生过早收敛等问题,体现在图像增强效果上即是图像明暗分配不当,灰度分布不均匀,观感可能比原图更差。故目前的研究思路多考虑将多种算法进行混合以平衡局部寻优和全局搜索能力来增强算法收敛性能,进而改善图像增强的效果。

相比于经典的遗传算法、人工免疫算法、粒子群算法、鱼群算法等,细菌觅食算法具有局部搜索能力强、收敛性能好、本质并行性等优点<sup>[5]</sup>,且算法结构简洁易于实现,前人经过大量的基准函数测试,结果表明其性能均大幅优于上述经典智能算法,故本文选用了细菌觅食算法来进行图像增强处理。

然而细菌觅食算法仍存在改进的空间,如经典的 BFO 算法有三层嵌套循环,导致计算复杂度过高,在优化高维图像问题上易陷入局部最优的缺点。目前的改进思路一般有算子参数设置优化和混合算法改进两个方面,相关成果归纳如下:

### 1) 算法本身的改进

Dasgupta 和 Das 等人从理论上分析了自适应移动步长对算法收敛性与稳定性的影响,提出了将固定趋化步长改进为自适应步长的思想;Liu Y 等初步分析了算法的收敛性,并对大肠杆菌间的相互作用机制进行了改进;Mishra 等基于模糊推理机制提出了最优步长选取方法<sup>[28][29]</sup>;刘小龙提出了基于高斯分布估计的细菌觅食优化算法;陈瀚宁等提出了自适应游动步长策略,并设计了一种协同细菌觅食算法<sup>[26]</sup>;姜建国、周佳薇等提出了变概率的迁徙机制,以解决“精英逃逸”现象。

### 2) 混合算法研究

刘小龙等提出了将分布估计和差分进化思想引入繁殖操作<sup>[4][6]</sup>;任佳星等结合遗传算法在菌群进化过程中引入了交叉和变异算子;Sun 等提出了具有量子行为和高斯分布的改进 BFO<sup>[27]</sup>;储颖等提出了一种基于粒子群优化的快速细菌算法。

以上改进算法借鉴了遗传算法、粒子群算法、混沌优化的思想对细菌觅食算法的几个算子进行了混合研究并取得了良好的效果,但也存在考虑不全面的问题,如引入交叉和变异算子易导致群体中优秀个体的缺失;刘小龙的分布估计虽然能够提高搜索精度和速度,但在优化 Schwefel 等极易陷入早熟收敛的函数时效果不理想。

综上,由于细菌觅食算法提出时间较晚,算法理论层面尚未成熟,还有很大的改进空间。且其在图像处理领域的应用先例并不算多,因此对其算法理论本身的改进和在图像处理及其他工程领域的应用研究亟待展开。

### 1.3 本文工作

本课题的研究旨在将基于生物启发的细菌觅食优化算法引入到图像空域增强技术中，以增加传统图像增强方法过程中参数选取的智能性，并针对经典 BFO 存在的问题进行改进研究，以提升图像处理的性能和优化效果。本文的主要研究内容如下：

#### 1) 对传统的图像增强方法做了简要介绍

传统的图像增强方法有空域法、频域法、模糊法三大类，重点研究了简单高效的空域法。对图像的点运算，如线性变换、分段线性变换、非线性变换、直方图均衡化，以及空域运算中的线性滤波、平滑、锐化等方法均作了简明论述，这些方法是后续研究的基础，也因其实时计算速度快的优点有其适用的领域，可与智能优化方法互补长短。

#### 2) 研究了经典 BFO 算法的优化原理和算法流程

广泛研究了这个领域的研究现状及 BFO 算法的相关理论，分析了经典 BFO 存在的不足。细菌觅食算法通过细菌个体间的竞争和协作完成全局寻优，系统初始化为一组随机解，每个细菌的位置代表一个候选解，通过趋化操作实现位置更新，繁殖和消亡的过程体现了适者生存的思想，迁徙过程是为了防止菌群陷入局部搜索停滞现象。三个步骤嵌套执行，最终得出一个最优的目标函数适应度值。

#### 3) 对经典 BFO 算法提出了改进策略并验证了有效性

针对经典细菌觅食优化算法全局寻优能力较差、在高维多模态问题上易陷入早熟收敛、“精英逃逸”现象等缺点，在经典 BFO 的基础上结合其他群智能算法的优点对其进行了改进研究，提出了一种改进算法——具有协同思想的自适应迁徙概率细菌觅食算法。

针对经典 BFO 四个操作过程（种群初始化、趋化、复制、迁徙）的优化机制分析，采用了随机性佳点集方法生成初始种群以增加解空间的多样性；引入聚集算子，添加其他细菌对每个个体的影响；引入粒子群算法中的个体认知和社会学习思想，提出了变权重系数的协同思想趋化方向计算策略；结合变次数游动策略提高了解的精度和搜索效率；采用自适应迁徙概率方式解决细菌寻优过程中“精英逃逸”现象，加快了算法收敛速度，避免早熟收敛问题。并用改进后的算法对几个经典的基准函数进行了 MATLAB 仿真测试实验，结果表明改进算法在全局收敛、寻优精度和速度方面都有很大的性能提升。

#### 4) 将改进后算法应用于图像空域增强问题

针对图像本身的高维特点，变换思路，将变高维的灰度图像增强问题转换为固定二维的不完全 beta 函数参数寻优问题，拟合计算最优的非线性灰度变换曲线，设计了能全面兼顾图像质量评价指标的目标函数以衡量算法性能。仿真实验结果表明了本文方法的有效性。

对于改进后的细菌觅食优化算法，如何合理选择和调整控制参数是影响算法性能的关键，并在高维函数寻优问题上还存在求解精度的不足。后续工作将继续探寻算法参数设置的规律，以改善算法性能。

### 1.4 内容安排

本文共分为五章，各章节的主要内容安排如下：

第一章——绪论。阐述了论文的研究背景，综述了图像空域增强问题以及群智能算法的研究现状，并指出了对细菌觅食算法的研究改进的必要性，最后总结了论文和后续工作安排。

第二章——传统图像增强方法概述。阐述了图像增强处理的相关知识的定义和概念，介绍了一些常用的图像增强方法，并给出了 MATLAB 仿真效果的比较和分析，讨论了传统增强方法的优缺点及适用领域，为后文的群智能优化方法处理做好知识铺垫。

第三章——经典细菌觅食算法研究。阐述了群智能优化算法和经典 BFO 的定义及相关概念，分析了经典 BFO 的优化流程和寻优步骤，讨论了参数设置对算法性能的影响，指出了其不足之处，为后续改进准备了基础。

第四章——细菌觅食算法改进研究及仿真测试。针对经典 BFO 中的缺点进行了改进，引入了粒子群算法中的个体认知和社会学习思想，并设计了自适应概率的迁徙策略，对参数设置进行了比较优化，最后，通过仿真实验验证了改进后 BFO 算法的高效性。

第五章——改进算法应用于灰度图像增强问题。设计了全面兼顾图像质量评价标准的适应度函数作为算法的寻优目标函数，将变高维的图像增强问题转化为固定二维的不完全 beta 函数参数寻优问题，选取了几张经典的降质图片分别用传统图像增强方法、经典 BFO 和改进 BFO 进行处理，通过对比分析验证了本文提出的改进算法的有效性。

结论和展望。本章总结全文内容，对后续应用研究工作进行了展望。



## 第 2 章 图像增强方法概述

一个数字图像处理系统的处理流程一般可分为 3 个阶段——预处理阶段、特征提取阶段、识别分析阶段。其中，预处理阶段包括图像增强、阈值分割等，是后续工作的基础。

本章将对传统的图像增强方法作概要介绍，包括图像的像素点运算、空域增强、频域增强、模糊增强方法等，基本涵盖了目前图像增强处理中常用到的所有典型方法。需要注意的是，目前图像增强领域还没有一个适用于所有问题的通用算法，即使是用群智能算法来处理，虽能取得较好的增强效果，但是如果用在需要实时计算的领域的话运行速度是令人难以接受的。所以传统的图像增强方法也是有学习研究的必要，在实际工程领域中往往是对多种方法的组合使用，本章介绍的处理算法也是后续做群智能算法应用的基础。

### 2.1 图像的像素点运算

像素点运算是对于图像中的每个像素直接执行同样的变换操作。设 $r$ 和 $s$ 分别是输入图像 $f(x, y)$ 和输出图像 $g(x, y)$ 在一点 $(x, y)$ 的灰度值，像素点运算的一般形式如下：

$$s = T(r) \quad (2.1)$$

其中， $T$ 为的像素点运算算子，表示某种灰度级映射关系。像素点运算的主要作用是用来改变图像的灰度范围及分布，如对图像的对比度进行增强、拉伸或进行灰度变换等操作，下面介绍常用的变换方法。

#### 2.1.1 灰度直方图

灰度直方图统计了一幅图像中的各灰度级像素点出现的次数或频率，是最基本的图像处理分析工具，主要用于图像灰度变换、阈值分割等处理过程中<sup>[11]</sup>。

其横坐标是像素灰度级，一般灰度图像的灰度范围是 0~255，纵坐标是个数或频率，数学上表示了图像灰度值函数的关系，也有一种的特殊的归一化直方图，可以直接反映不同灰度级出现的比率。

在 MATLAB 中可以很方便的调用 `imhist()` 函数进行图像的灰度直方图运算。下图显示了 `lena` 图像的直方图信息：

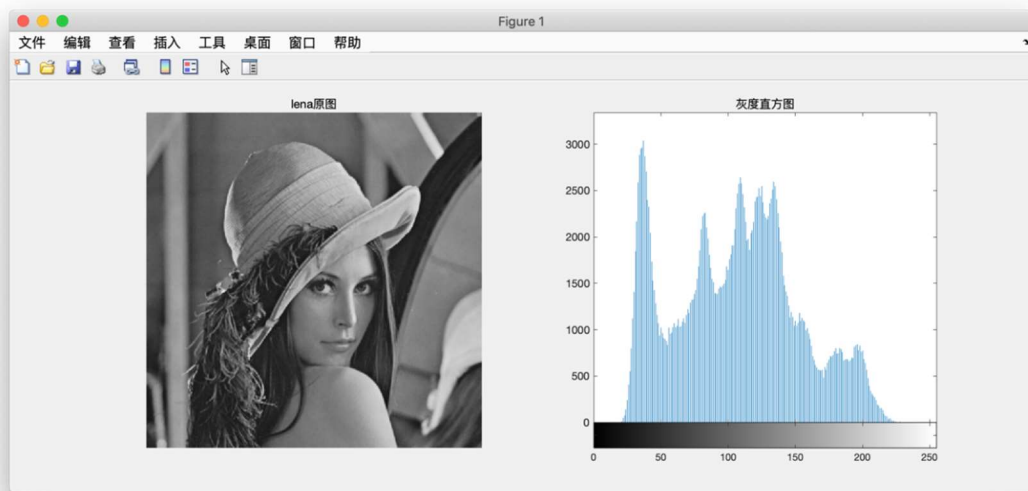


图 2-1 图像直方图

`imhist()`函数的返回值中包含了落入每个灰度级的像素点个数，与图像像素点总数做商运算可得到归一化的灰度分布直方图，效果如下所示：

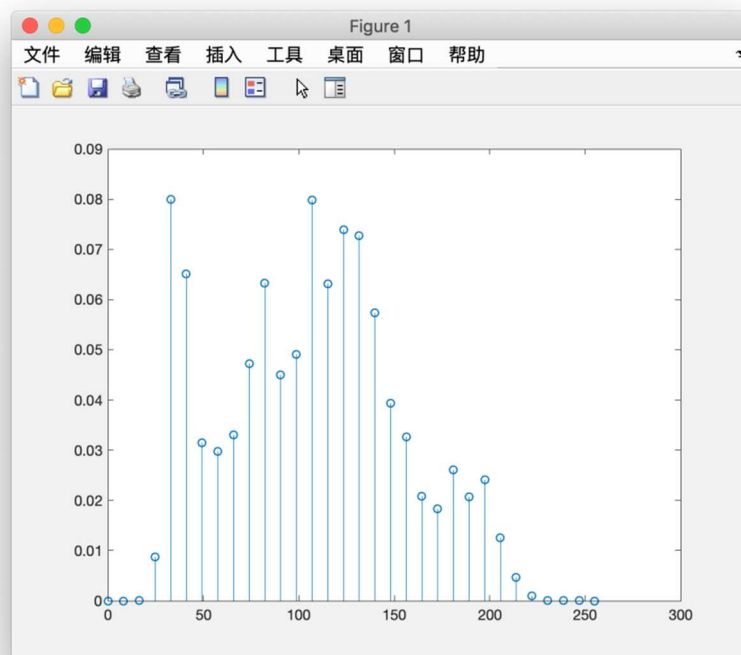


图 2-2 归一化直方图

分析图像的直方图往往可以得到很多有效的信息。比如，直方图中灰度峰值出现的位置反映了图像的亮暗信息：峰值靠左，则图像较暗；峰值靠右，则图像较亮；灰度分

布只集中在一个狭窄的灰度级区域，那么图像对比度就比较低；反之，灰度分布宽广均匀，那么说明图像对比度就越好。

### 2.1.2 线性变换和分段线性变换

灰度线性变换函数 $f(x)$ 是一个一维线性函数：

$$B = f(A) = f_A A + B \quad (2.2)$$

其中， $f_A$ 为函数的斜率， $B$ 为函数在 $y$ 轴上的截距， $A$ 表示输入图像灰度， $B$ 表示输出图像灰度。斜率 $f_A$ 的不同取值，可对输入图像施加不同的操作：

- 当 $f_A > 1$ 时，会增大图像对比度；当 $f_A < 1$ 时，会减小图像对比度。
- 当 $f_A = 1$ 且 $B \neq 0$ 时，使得所有的像素的灰度值上移或下移，其效果是整个图像将变得更暗或更亮；如 $f_A < 0$ ，将变亮暗区域，变暗亮区域。如果线性变换操作超过了图像灰度级范围（如小于 0 或大于 255）将使亮度饱和而丢失部分细节。
- 当 $f_A = 1, f_B = 0$ 时，输出图像即原图；当 $f_A = -1, f_B = 255$ 时，则会反转原图灰度。

MATLAB 中对图像作线性变换直接设计线性函数即可，无需专门函数，下图是对 coins 图像进行不同参数的线性变换效果，并显示了变换后各自的灰度分布直方图：

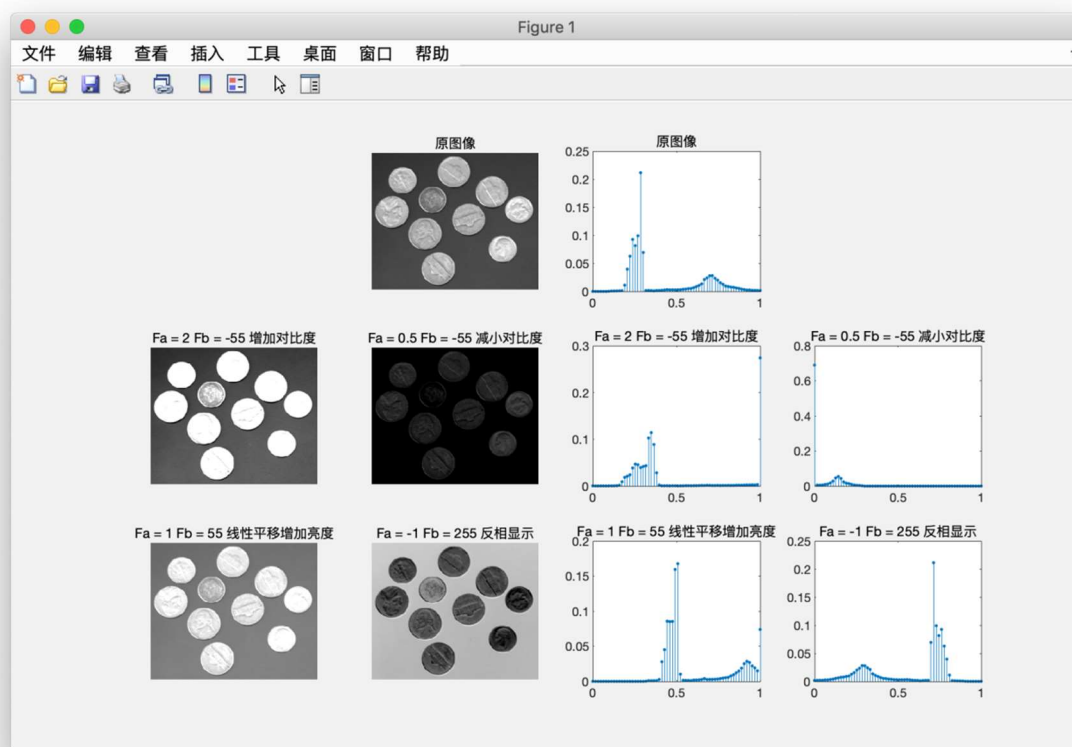


图 2-3 灰度线性变换

从灰度直方图可以看出，对直方图的缩放和平移可以改变图像对比度；平移直方图在横轴上的位置可以改变亮度，左移则变暗，右移则变亮；对直方图水平镜像操作可使图像灰度反转。简单的线性变换可以一定程度上改善图像视觉对比度，但是缺乏对图像细节进行增强的能力，一般将线性变换和非线性变换结合运用。

分段线性变换则是对原灰度图不同的灰度范围执行不同的变换操作，这样就可只增强感兴趣的灰度范围，而抑制不感兴趣的部分。分段线性变换的优点在于可以灵活地对任意部分灰度范围执行不同的变换操作，缺点是用户需要考虑的较多。

分段线性变换的形式如下：

$$f(x) = \begin{cases} \frac{y_1}{x_2} & x < x_1 \\ \frac{y_2 - y_1}{x_2 - x_1}(x - x_1) + y_1 & x_1 \leq x \leq x_2 \\ \frac{255 - y_2}{255 - x_2}(x - x_2) + y_2 & x_2 < x \end{cases} \quad (2.3)$$

其中， $x_1$ 和 $x_2$ 为分段的灰度范围， $y_1$ 和 $y_2$ 是线性函数斜率。

### 2.1.3 对数变换

对数变换的一般表达式如下：

$$t = a \log(1 + r) \quad (2.4)$$

其中， $a$ 为尺度比例常数， $r$ 为源灰度值， $t$ 为变换后灰度值。对数变换示意图如下：

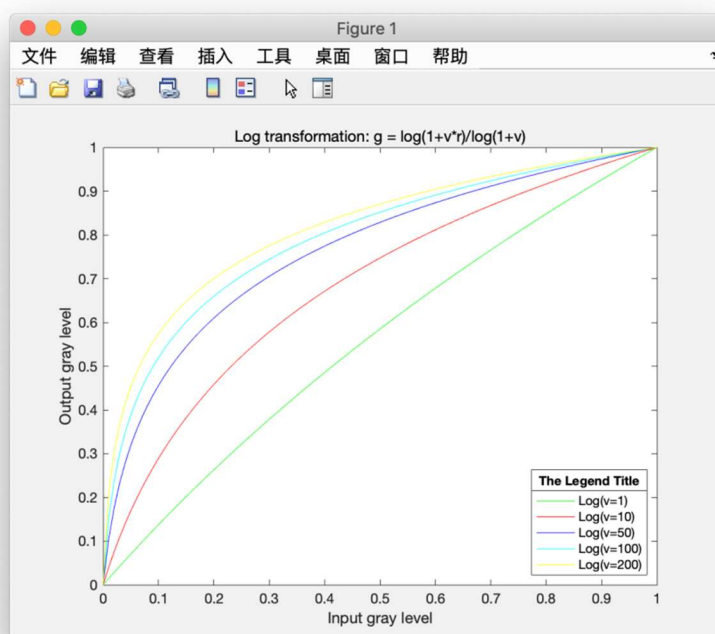


图 2-4 对数变换

上图的对数变换曲线中，函数自变量的值较低时，曲线斜率较高；函数自变量的值较高时，曲线则趋向平滑，用对数函数来作灰度变换函数时，对应的变换效果就是可扩展被压缩的低值灰度区间，而不对高值部分产生影响。这一特性可以被用在改善频谱图像的显示中。一般的显示器动态显示范围不够，就会丢失大量暗部细节，通过对数变换扩展暗部灰度区间，则可以清晰显示更多的信息。

下面对傅里叶频谱图像作灰度对数变换的效果如下：

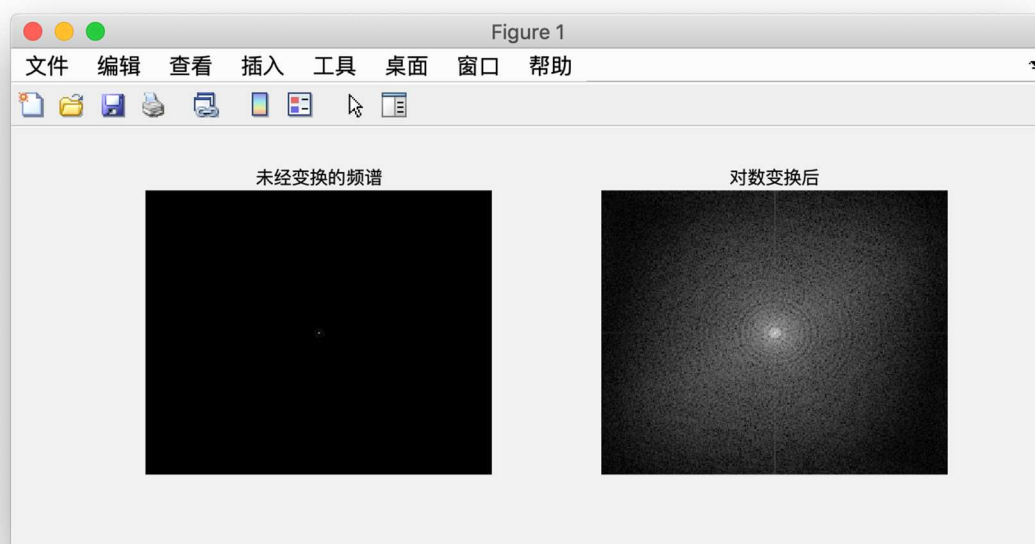


图 2-5 傅里叶频谱图像

从两幅图像的对比中可以清楚看到经过对数变换后的频谱图像因为扩展了暗部灰度区域，可以清晰显示细节，而原图中因为绝对高灰度值的存在压缩了低值部分，只可辨认出中心一点。

#### 2.1.4 伽玛变换

伽玛变换是另一种常用的非线性灰度变换，也叫指数变换、幂次变换。其表达式如下：

$$y = (x + esp)^\gamma \quad (2.5)$$

其中， $x$ 与 $y$ 的取值范围均为 $[0,1]$ ， $esp$ 为补偿系数， $\gamma$ 为伽玛系数。 $\gamma$ 的取值表示灰度映射方式，三种取值情况如下：

- $\gamma > 1$ 时，增强图像高灰度区域对比度。
- $\gamma < 1$ 时，增强图像低灰度区域对比度。
- $\gamma = 1$ 时，不改变原图，线性输出。

在执行伽玛变换操作时，需要先对图像做归一化处理，即将原图的灰度区间由 $[0,255]$ 变换到 $[0,1]$ 区间，处理完成后再反归一化操作至 $[0,255]$ 区间，得到增强后图像。

MATLAB 中可以调用 `imadjust()` 函数来实现伽玛变换，其调用语法如下：

```
J=imadjust(I,[low_in,high_in],[low_out,high_out],gamma);
```

下面给出了不同 `gamma` 取值时的伽玛变换效果：

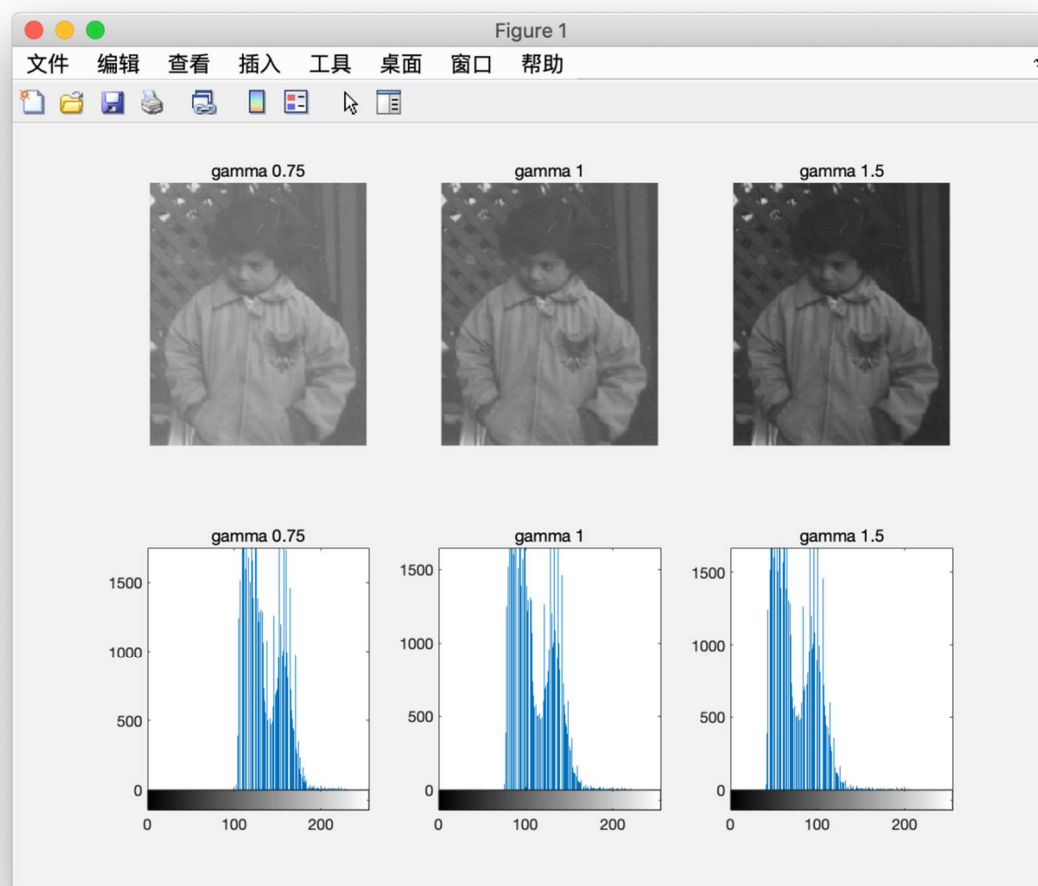


图 2-6 伽玛变换

可以看到，不同 `gamma` 取值会对直方图中非零区间的位置产生改变。伽玛变换属于非线性变换，不仅可以改善图像的对比度，亦可对图像的细节进行增强处理，改善整体图像视觉效果。

### 2.1.5 直方图均衡化

直方图均衡化是通过某种灰度映射将输入图像转换为在各个灰度级上分布均匀的输出图像，使得图像具有较高的对比度和较大的动态范围，灰度色调更丰富，从而易于进行判读<sup>[16]</sup>。下面对直方图均衡化变换过程的数学基础进行推导。

当图像的灰度范围为 $[0,1]$ 区间且连续时，图像的归一化直方图即为概率密度函数。

$$p(x), 0 \leq x \leq 1 \quad (2.6)$$

做积分操作，则有以下关系：

$$\int_{x=0}^1 p(x)dx = 1 \quad (2.7)$$

设原图概率密度函数为 $p_r(r)$ ，输出图像概率密度函数为 $p_s(s)$ ，变换函数为 $s = f(r)$ 。则有以下式：

$$p_s(s) = p_r(r) * \frac{dr}{ds} \quad (2.8)$$

要想使输出图像的概率密度函数： $p_s(S) = 1, 0 \leq S \leq 1$ （即输出直方图是均衡化的），则有以下式：

$$p_r(r) = \frac{ds}{dr} \quad (2.9)$$

两边对 $r$ 做积分操作，可得下式：

$$s = f(r) = \int_0^r p_r(\mu)d\mu \quad (2.10)$$

式(2.10)即为图像的累积分布函数。反归一化到 $[0,255]$ 区间，乘以最大灰度值 $L_{max}$ 得到均衡化后图像，此时的变换公式如下：

$$B = f(A) = L_{max} \int_0^A p_A(\mu)d\mu \quad (2.11)$$

其中， $B$ 为输出图像灰度值， $A$ 为原图灰度值。

对于离散灰度级则有如下变换公式：

$$B = f(A) = \frac{L_{max}}{A_0} \sum_{i=0}^A H_i \quad (2.12)$$

其中， $H_i$ 为第 $i$ 级灰度像素个数， $A_0$ 为像素总数，即图像面积。

MATLAB 中提供了用于直方图均衡化的函数 `histeq()`，调用语法如下：

`[J,T]=histeq(I);` %  $I$  是输入图像， $J$  是输出图像， $T$  是变换矩阵

可利用直方图均衡化技术来实现图像的灰度归一化来消除灰度因素（如光照、视角、方位、噪声等）造成的图像外观变化。下面的程序对输入图像分别进行了增加对比度、减小对比度、线性增加亮度和线性减小亮度等处理，再对其进行直方图均衡化处理，得到了处理前后直方图的变化对比结果：



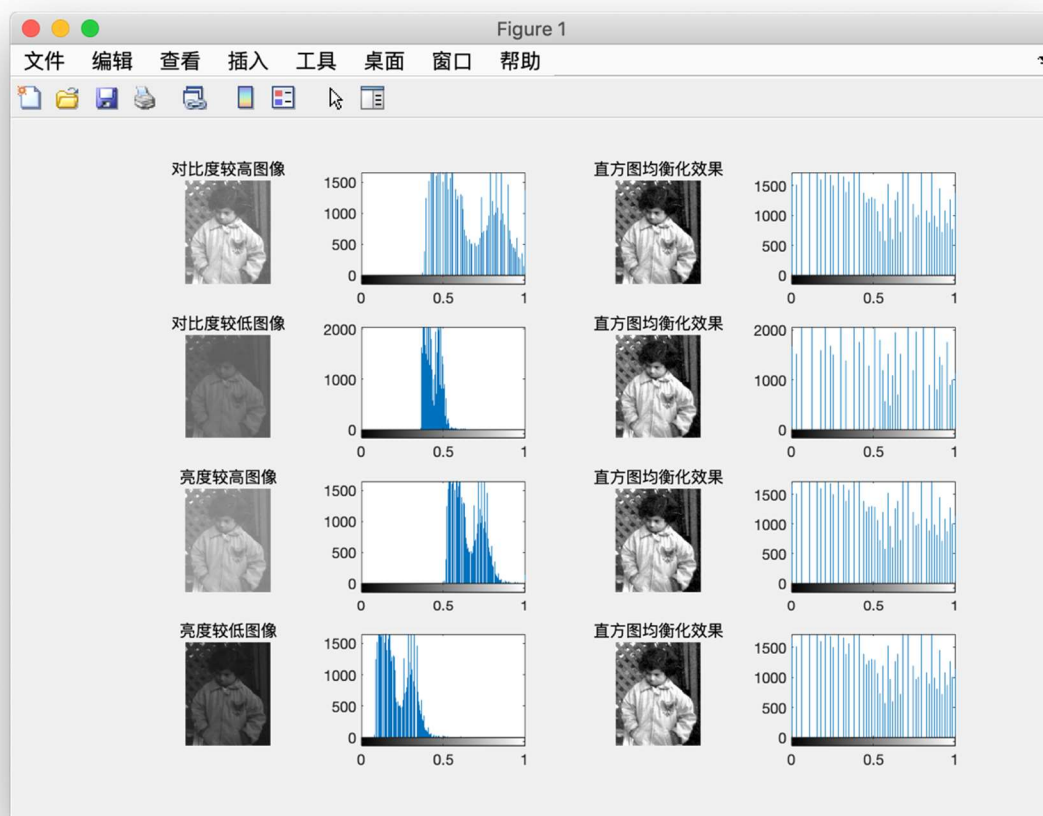


图 2-7 不同亮度和对比度图像直方图均衡化后效果

可以看到，对于不同亮度和对比度的图像，经直方图均衡化处理后的视觉效果和直方图分布几乎完全一致，这说明在进行图像分析和比较之前将图像转化为统一的形式是十分有用的。

## 2.2 空间域图像增强

图像空域增强方法有直方图均衡化、图像平滑、图像锐化等，都是建立在灰度变换的基础之上<sup>[21]</sup>。一般单一的处理方式并不能得到很好的增强效果，往往需要组合运用，才能获得满足需求的增强图像。上一节介绍的灰度变换和直方图灰度修正技术都是图像增强的有效手段，这些方法的共同点是变换直接针对像素值，而与像素所处区域无关。空域增强则是基于图像中每一个小范围（邻域）内的像素进行变换操作，每个点的变换效果由该点领域内所有像素点共同决定，故空域增强也叫领域运算或邻域滤波。空域变换可由下式描述：

$$g(x,y) = T[f(x,y)] \quad (2.13)$$



### 2.2.1 空域滤波

空域滤波主要是直接基于邻域对图像像素执行计算<sup>[24]</sup>。对图像中的每个像素点 $(x, y)$ ，执行如下操作：

- (1) 将以 $(x, y)$ 为中心的邻域内的像素作为模板进行变换处理；
- (2) 将(1)中运算结果作为 $(x, y)$ 点新的响应。

如果对于邻域内的像素处理为线性运算，则称线性空域滤波，否则为非线性空域滤波。选取的邻域模板也称为滤波器、掩膜、窗口等<sup>[67]</sup>。滤波过程即是在原图像中逐个像素点移动模板的过程，对所有点执行预先定义的模板变换操作后即得到增强后图像。

执行滤波操作时需要注意当模板处于图像边缘时，会出现模板的某些元素位于图像之外的情况，这时要制定响应的边界处理策略。通常使用以下3种策略来解决边界问题：

- (1) 收缩处理范围。即忽略位于图像边界附近会引起问题的点。
- (2) 使用常数填充。根据模板形状为超出边界的部分填充指定的常数，如0或255。
- (3) 使用复制像素填充。与(2)类似，只是填充的值是对图像边界的复制。

MATLAB 中滤波函数有 `imfilter()` 和 `fspecial()`，前者执行滤波操作，后者可创建预定义的二维滤波器。下面给出了对 `cameraman` 图像进行相关滤波操作的效果，采用的是复制边界的填充方式：



图 2-8 空域滤波效果图

## 2.2.2 图像平滑

图像平滑可用来减少和抑制图像噪声，在空间域中一般使用邻域平均来实现。下图中所使用的就是一个  $3 \times 3$  的平均模板，在中心像素点周边  $3 \times 3$  邻域内的点都参与了新图像该中心点像素值的运算，系数表示邻域内每个点的权重。

不同大小的平均模板对图像平滑效果的影响如下所示：



图 2-9 平均平滑效果

从 baby 图像的平均平滑的效果图可以看出，模板越大，则滤除的噪声越多，但是同时图像也会变得越来越模糊，这是平均平滑的工作原理使然。所以我们在使用平均平滑方式时需要事先预估噪点的大小，再有针对性的选取模板。

平均平滑的处理效果往往会模糊且不自然，为规避这一缺点就需要用到高斯平滑，其思想即来源于高斯分布函数，邻域模板点的权重系数随着与中心像素点的距离增大而迅速减小，这样离中心点越近的点所施加的影响也就越大。常用的  $3 \times 3$  高斯模板如下：

$$w = \frac{1}{16} \times \begin{pmatrix} 1 & 2 & 2 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix} \quad (2.15)$$

高斯平滑的处理由离散化的二维高斯函数确定：

$$M(i,j) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{((i-k-1)^2 + (j-k-1)^2)}{2\sigma^2}\right) \quad (2.16)$$

$\sigma$ 取值不同对二维高斯函数的形状影响很大，取值过小则加权和响应基本不考虑邻域影响，退化为点运算；取值过大则退化为平均模板。一般在 MATLAB 中对于  $3 \times 3$  的模板取  $\sigma$  为 0.8 左右，对于更大的模板可适当增大  $\sigma$  的值。



图 2-10 高斯平滑效果

随着模板的增大，原图中的噪声得到了很好的抑制，对比平均平滑的结果，同样是  $7 \times 7$  的模板大小，高斯滤波后的图像中图像细节被较好地保留。

平均平滑和高斯平滑都属于线性平滑滤波器，在频率域滤波中也叫低通滤波器。

### 2.2.3 中值滤波

中值滤波是对中心像素点邻域模板内所有像素的统计排序中值作为该点新的响应，所以其本质上是一种统计排序滤波器，故其是非线性滤波器。

中值滤波因为只选取排序中值而完全舍弃掉邻域模板内其他像素点的特性，对于一些随机类型的噪声，如椒盐噪声，有很好的降噪能力，相比之下，线性平滑滤波要把阶跃较大的噪点都计算在内，往往去噪效果不佳。

MATLAB 中使用 `medfilt2()` 函数来实现中值滤波，下面对椒盐噪声的平滑效果进行比较：



图 2-11 中值滤波效果

可见线性平滑滤波在降噪的同时也不可避免地造成了模糊，而中值滤波在有效抑制椒盐噪声的同时模糊效应要低得多，因此对于椒盐噪声污染的图像，中值滤波效果要远优于线性平滑滤波。

此外，作为一种非线性滤波，中值滤波有可能会改变图像的性质，故一般不适用于军事图像和医学图像处理等领域。

## 2.2.4 图像锐化

图像锐化操作可使模糊的图像变得更清晰，在医学成像、工业检测、军事系统等领域应用广泛。

图像平滑去除噪声做的是抑制灰度跳变操作，而锐化需要突出边缘，做的是增强灰度跳变操作。所以两者互为逆运算，在数学基础上也是如此，如平滑往往是做加权求和、积分运算，锐化操作通常用求导（梯度）和有限差分来实现。

### 基于一阶导的图像增强——梯度算子

二维连续函数 $f(x, y)$ 在点 $(x, y)$ 处的梯度是如下的二维列向量：

$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \quad (2.17)$$

其中， $\frac{\partial f}{\partial x}$ 是在点 $f(x, y)$ 处 $f$ 对 $x$ 的偏导； $\frac{\partial f}{\partial y}$ 是在点 $f(x, y)$ 处 $f$ 对 $y$ 的偏导。梯度的方向即是函数 $f(x, y)$ 变化率最大的方向。梯度的幅值是变化率大小的度量，其值为：

$$|\nabla f(x, y)| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}.$$

二维离散函数 $f(i, j)$ 的梯度幅值可以用有限差分来近似，如下式所示：

$$|\nabla f(i, j)| = \sqrt{[f(i+1, j) - f(i, j)]^2 + [f(i, j+1) - f(i, j)]^2} \quad (2.18)$$

上式中的平方和开方增加了计算复杂度，所以可以绝对值形式近似表示：

$$|\nabla f(i, j)| = |f(i+1, j) - f(i, j)| + |f(i, j+1) - f(i, j)| \quad (2.19)$$

实际应用中往往使用如下的 Robert 交叉梯度来近似代替上式：

$$|\nabla f(i, j)| = |f(i+1, j+1) - f(i, j)| + |f(i, j+1) - f(i+1, j)| \quad (2.20)$$

Robert 交叉梯度对应的模板如下：

$$w1 = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \quad w2 = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

其中， $w1$ 可对接近  $45^\circ$  边缘响应较好，而 $w2$ 对接近 $-45^\circ$  边缘响应较好。

在 MATLAB 中进行 Robert 交叉梯度锐化的效果如下：

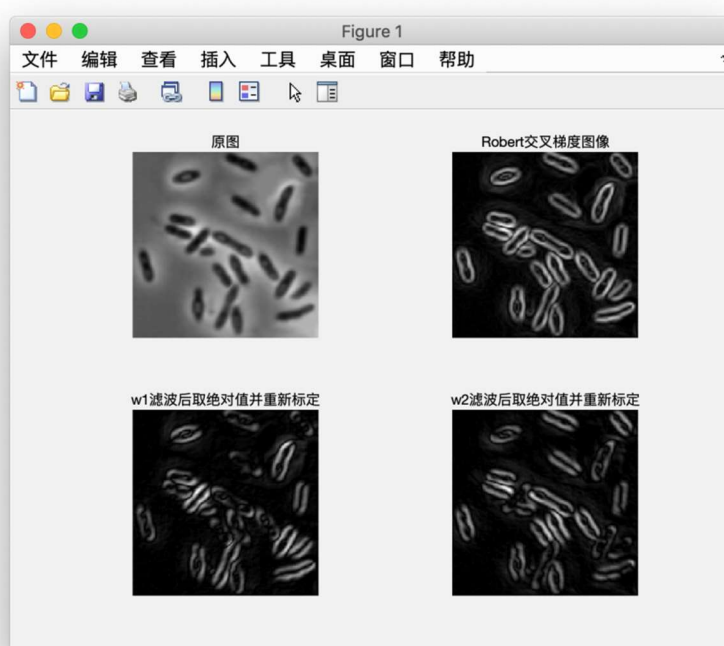


图 2-12 Robert 算子锐化效果

实际滤波时使用奇数尺寸模板效果更好, 所以有如下计算 Sobel 梯度的 Sobel 模板:

$$w1 = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad w2 = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad (2.21)$$

下面的 MATLAB 程序计算了一幅图像的竖直和水平梯度, 两者之和可作为完整的 Sobel 梯度。也可使用 `gradient()` 函数来计算 Sobel 梯度。

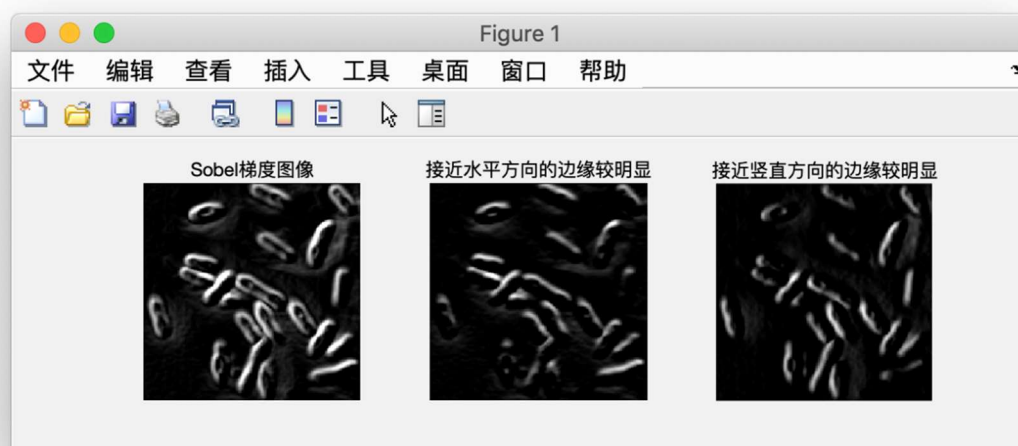


图 2-13 Sobel 算子锐化效果

### 基于二阶微分的图像增强——拉普拉斯算子

基于二阶微分的拉普拉斯（Laplacian）算子处理效果较一阶导的梯度算子更优，应用也更为广泛。对于二维函数 $f(x, y)$ ，其定义为：

$$\nabla^2 f(x, y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad (2.22)$$

图像通常是离散信息，所以对于二维离散图像 $f(i, j)$ ，有下式来近似二阶偏微分：

$$\begin{aligned} \frac{\partial^2 f}{\partial x^2} &= (f(i+1, j) - f(i, j)) - (f(i, j) - f(i-1, j)) = f(i+1, j) + f(i-1, j) - 2f(i, j) \\ \frac{\partial^2 f}{\partial y^2} &= (f(i, j+1) - f(i, j)) - (f(i, j) - f(i, j-1)) = f(i, j+1) + f(i, j-1) - 2f(i, j) \end{aligned}$$

以上二式相加即可得到拉普拉斯算子：

$$\nabla^2 f = [f(i+1, j) + f(i-1, j) + f(i, j+1) + f(i, j-1)] - 4f(i, j) \quad (2.23)$$

对应的滤波模板如下：

$$w1 = \begin{bmatrix} 0 & 1 & 1 \\ 1 & -4 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

在锐化增强中，绝对值相同表示的响应也相同，故也等同于如下模板  $w2$ ：

$$w2 = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

拉普拉斯算子模板的结构决定了其对于水平和竖直方向的边缘都有很好的增强效果，即  $90^\circ$  各向同性，那么同样可设计如下模板使其对于  $45^\circ$  的旋转各向同性：

$$w3 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad w4 = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

借鉴高斯模板的思想，按照距离中心点的远近取不同的权重系数，可得到如下模板  $w5$ ：

$$w5 = \begin{bmatrix} 1 & 4 & 1 \\ 4 & -20 & 4 \\ 1 & 4 & 1 \end{bmatrix}$$

使用上述 3 种拉普拉斯模板的滤波程序运行结果如下：



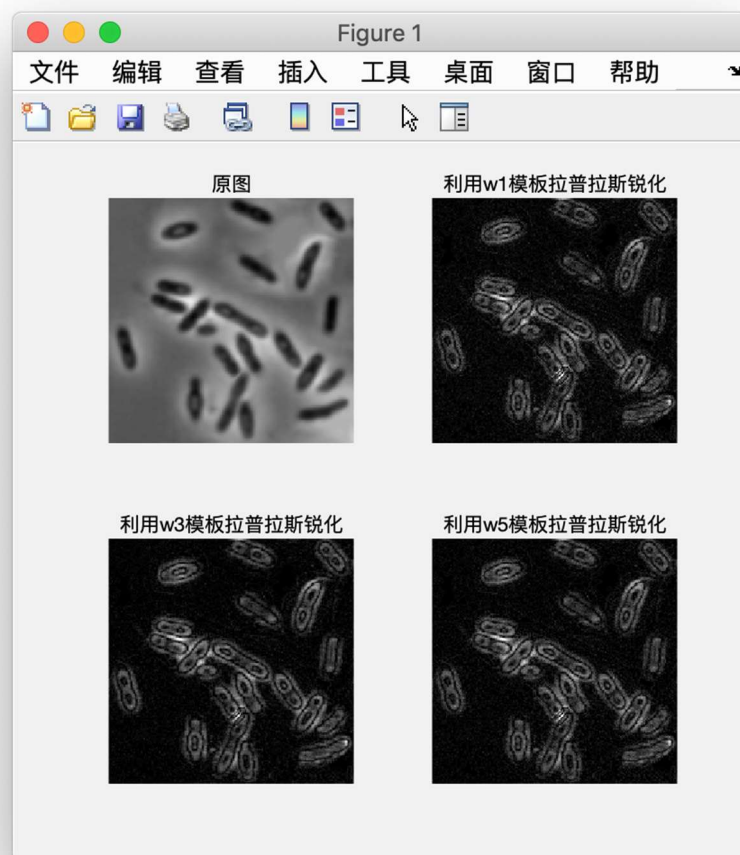


图 2-14 拉普拉斯锐化效果

从上面的运行效果图可以看出，拉普拉斯算子锐化处理的图像会有双边缘出现，这一点在之前的 Robert 算子和 Sobel 算子中是没有的；此外，也可以看出，拉普拉斯算子处理还会增强一些离散的噪点，这一性质需要通过下面介绍的高斯-拉普拉斯变换来规避。

### 高斯-拉普拉斯变换（Laplacian of Gaussian, LoG）

如何区分噪声和边缘信息是锐化处理中需要解决的一个核心问题，因为使用一般的拉普拉斯锐化算子锐化的同时，也会对噪点进行了增强。想要在增强图像细节的同时将噪点的影响去除，可以先对图像进行平滑操作，再对平滑后的图像进行锐化操作，如此便得到了高斯-拉普拉斯算子（由 Marr 和 Hildreth 提出）。理论基础如下。

对于高斯型函数：

$$h(r) = -e^{-\frac{r^2}{2\sigma^2}} \quad (2.24)$$

其中， $r^2 = x^2 + y^2$ ， $\sigma$  为标准差。 $\sigma$  决定平滑程度，使用高斯函数进行滤波便可得到高斯平滑图像。计算  $h$  的拉普拉斯算子（ $h$  关于  $r$  求二阶导数），可得到著名的高斯-拉普拉斯算子：



$$\nabla^2 h(r) = - \left[ \frac{r^2 - \sigma^2}{\sigma^4} \right] e^{-\frac{r^2}{2\sigma^2}} \quad (2.25)$$

下图为一个 LoG 函数的三维形状。将上式经过离散化可近似为一个  $5 \times 5$  的拉普拉斯模板。

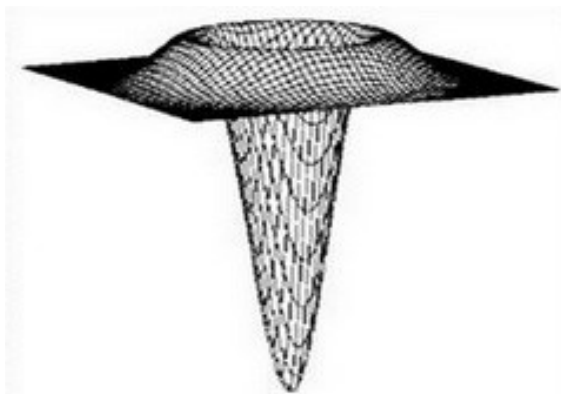


图 2-15 高斯-拉普拉斯函数的三维图形

下面给出了对于 lena 图像分别采用 Laplacian 和 LoG 算子进行锐化的 MATLAB 实现：

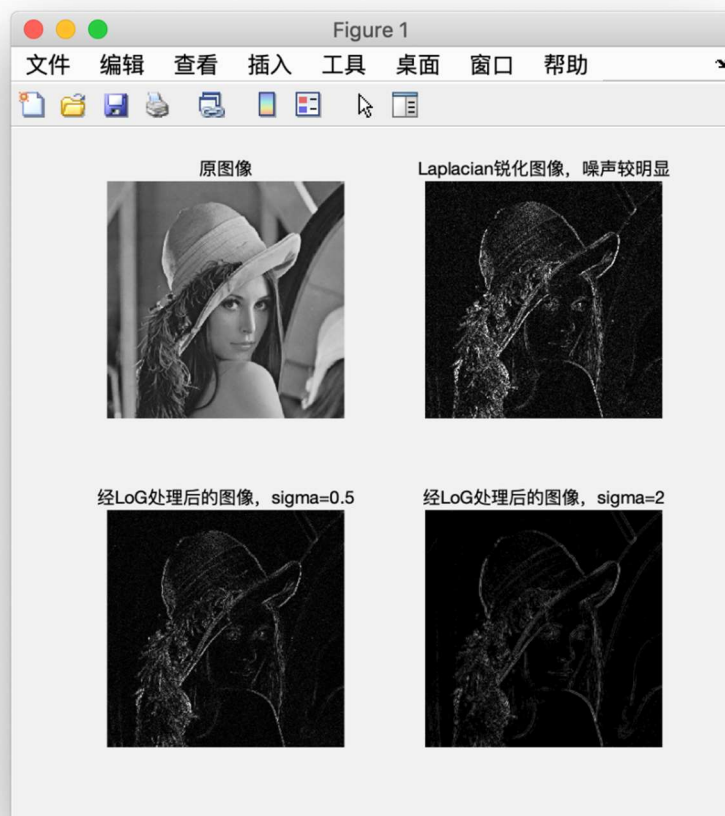


图 2-16 高斯-拉普拉斯变换效果

从运行结果可以看出，相比拉普拉斯锐化，经高斯-拉普拉斯算子处理后，有效抑制了噪声，且  $\sigma$  越小，细节增强效果越好， $\sigma$  越大则平滑效果越好。

### 2.3 频率域增强

频域增强的一般步骤是先将图像由空间域转换到频率域，如使用傅里叶变换，将一幅图像视为波，在频率域进行变换处理后再转换回空间域。其理论基础是卷积操作，卷积计算的公式如下式所示：

$$G(u, v) = H(u, v) * I(u, v) \quad (2.25)$$

再经逆变换得到增强后图像 $g(x, y)$ ：

$$g(x, y) = F^{-1}[G(u, v)] = F^{-1}[H(u, v) * F(u, v)] \quad (2.26)$$

其中， $F(x, y)$ 是原图的傅里叶变换， $H(x, y)$ 是滤波变换函数。

现有的频域增强方法有 Curvelet 和 Contourlet 变换等，这些方法在图像融合、图像分割和对比度增强方面有广泛的应用。频域增强也存在一些如傅里叶变换基函数的系数选取困难等缺点，在特定领域有很好的效果，但不如空域增强处理适用范围广，因本文研究课题的主要研究对象为空域增强方法，故这里不再赘述。

### 2.4 模糊增强

模糊增强处理方法是先将原始图像的邻域进行模糊化处理，变换为邻域中的模糊图片，执行相应变换后转换回空间域，得到增强效果<sup>[54]</sup>。模糊处理增强的流程一般分为三步：

- (1) 将图像从空间域转换到模糊域
- (2) 在模糊域进行增强处理
- (3) 从模糊域转换到空间域，得到增强后图像

一张大小为  $M \times N$ ，灰度级为  $L$  的图片可视为一个模糊点矩阵，记为  $X$ ：

$$X = \begin{bmatrix} p_{11}/x_{11} & p_{12}/x_{12} & \cdots & p_{1N}/x_{1N} \\ p_{21}/x_{21} & p_{22}/x_{22} & \cdots & p_{2N}/x_{2N} \\ \cdots & \cdots & \cdots & \cdots \\ p_{M1}/x_{M1} & p_{M2}/x_{M2} & \cdots & p_{MN}/x_{MN} \end{bmatrix}_{M \times N}$$

式中， $p_{ij}/x_{ij}$ 表示图像第 $(i, j)$ 个像素具有的特征度， $p_{ij}(0 < p_{ij} < 1)$ 。由点 $x_{ij}$ 变换到模糊域点 $p_{ij}$ 的过程称为图像模糊化，一般变换函数有标准模糊  $S$  函数、经典 PAL 函数、 $\sin$  函数等。

表 2-1 概括分析了空域法、频域法、模糊处理这三种增强方法的优缺点。

表 2-1 三类增强方法的比较

图像增强方法	优点	缺点
空域法	理论简单、复杂度低、 可广泛应用于实时图像处理	缺乏自适应能力
频域法	算法复杂度低，相应变换和特性的应用性好	不能满足对图像所 有部分都能很好地 处理
模糊处理	利用了图像本身的不确定性，符合人眼观察的习惯	需要人为操作确定 饱和点和渡越点、不 利于机器处理

## 2.5 本章小结

本章主要介绍了一些常用的传统空间域图像增强方法，包括图像像素点运算操作中的线性与非线性变换、直方图均衡化，以及空间域中的空域滤波概念，图像平滑、图像锐化等操作，并提出了对于受噪声污染的图像的处理方式往往是采取多种方法相结合的思路，着重介绍了锐化操作中的高斯-拉普拉斯变换，针对 MATLAB 仿真结果分析了各种方法中影响增强效果的因素。此外，对比分析了空域增强与频域增强、模糊增强的特性及优缺点，为后文应用群智能优化算法对图像空域增强做进一步的研究奠定了基础。

## 第3章 经典细菌觅食优化算法

细菌觅食优化算法是一种随机全局优化算法，为很多传统优化方法无法满足的实际工程问题指明了新方向，现已成功应用于电力系统调度、函数优化、神经网络等领域。本章将从细菌觅食算法的理论基础、基本概念、优化原理等方面对其进行详细介绍，给出经典 BFO 算法的具体流程，并对算法的优缺点进行深入分析，为后续章节的研究工作奠定基础。

### 3.1 群智能优化算法

实际生活需求促进了最优化方法的发展。近几十年来，由于传统优化方法在解决某些复杂实际工程问题时的不足和缺陷越来越突出，一些具有全局优化性能且通用性强的进化算法，因其高效的优化性能和无需精确描述问题信息等优点，受到各领域广泛的关注。其中产生最早也是最具代表性的进化算法是 70 年代源于达尔文自然选择学说和孟德尔遗传变异理论的遗传算法（GA）。

近十几年来，人们模拟自然界生物群体行为而产生的一系列群智能优化算法，如 Dorigo 等提出的蚁群算法，Eberhart 和 Kennedy 提出的粒子群优化算法，这些算法被广泛应用于工程领域并取得了显著的成果<sup>[8][10]</sup>。其是一种概率搜索算法，为解决无全局模型的复杂分布式问题提供了方案，其仿生物特性使得该类算法易于被转化为工程最优化问题，解决多目标优化、决策支持、模式识别、机器人控制等多个领域的最优化难题。

但目前学者对群智能优化的研究仍不够成熟，主要集中在对群体行为、理论基础的分析和控制方面，算法仍存在较多问题亟待解决。首先，该类算法的可行性与可靠性缺乏数学理论层面的基础证明；其次，该类算法仅针对某一特定问题提出，没有通用的算法结构；最后，种群之间的映射关系及种群中个体间的信息交互方式是群智能算法设计的关键，也是较难确立及标准化的问题。虽然群智能算法存在以上问题，但其对系统资源要求低、简单易实现的优点不容忽视，仍得到了广泛应用，其分布式、并行性和自学习的优点，为处理复杂工程最优化问题提供了技术保证。

### 3.2 细菌觅食算法原理

细菌觅食优化算法的思想来源于 Passino 对人类肠道中的大肠杆菌觅食行为的观察和实验，于 2002 年提出，2007 年引入国内<sup>[9]</sup>。目前，BFO 算法已被应用于函数优化分析、力学、作业调度、机器学习等问题方面。

大肠杆菌是现代医学上研究比较全面的一种微生物，其表面遍布着纤毛和鞭毛。在觅食过程中，靠纤毛和鞭毛的转动来向食物源方向移动，并能避开有毒物质。大肠杆菌的这种行为受控于自身控制系统，例如，当遇到酸性或碱性环境时，能够很好地避开，能够始终趋向于中性环境觅食，并在每一次改变状态之后及时对效果进行评价，为下一次状态的调整提供决策信息。

大肠杆菌的移动过程如下图所示，向前游动靠所有鞭毛都沿逆时针方向转动，而翻转方向靠所有鞭毛的顺时针转动。这种变化周而复始。在食物和外界环境适宜时，经过

一定时间后，将进行自我复制，使优秀细菌的数量加倍，而另外一些没有处于良好生存环境的大肠杆菌，则因没有足够的营养而消亡。由此整个菌群在避开有害物质，趋向有利环境的移动过程中实现了大肠杆菌群体的并行进化<sup>[21]</sup>。

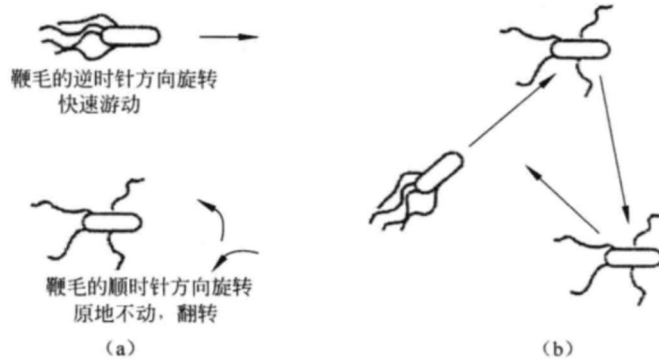


图 3-1 细菌觅食行为

### 3.2.1 趋化行为

趋化行为是细菌觅食机制的核心，趋化的目的是向着营养浓度高的区域聚集，具有局部寻优的特点。主要靠两个操作实现——翻转（Tumble）和游动（Swim）。翻转是为了找到一个适应度函数值好的方向，游动则是向着适应度值好的方向继续移动。细菌在开始觅食时，随机选择一个方向进行翻转，计算适应度函数值，若改善则沿着该方向游动一段距离，直到适应度函数值不再改善或达到最大游动步数为止；若适应度函数值未改善，则细菌再随机选择一个方向进行翻转，最差的情况是细菌一直在不断翻转寻找最优方向，计算适应度函数值。趋化操作通过翻转和游动反映细菌的探索和觅食行为，实现位置更新。

为了模拟实际细菌的行为，首先引入符号说明：

$j$  表示趋向性操作， $k$  表示繁殖操作， $l$  表示迁徙操作， $p$  表示搜索空间的维度， $S$  为细菌种群大小， $N_c$  为趋化次数， $N_s$  为趋化中在一个方向上游动的最大步数， $N_{re}$  为繁殖次数， $N_{ed}$  为迁徙操作次数， $P_{ed}$  为迁徙概率， $C(i)$  为细菌  $i$  的游动步长。

BFO 算法中先随机初始化一组菌群：

$$X = X_{min} + rand * (X_{max} - X_{min}) \quad (3.1)$$

其中， $X_{\{max\}}$  和  $X_{min}$  是优化区间的最大值和最小值， $X$  为细菌初始化位置。

设  $P(i, j, k, l)$  为细菌个体  $i$  在第  $j$  次趋向操作、第  $k$  次繁殖操作、第  $l$  次迁徙操作后的位置， $\phi(i, j)$  为细菌  $i$  翻转时的单位随机方向向量， $\Delta(i)$  为随机方向向量，其值介于  $[-1, 1]$ 。则每次趋化操作后细菌的位置按如下公式更新：

$$P(i, j, k, l) = P(i, j, k, l) + C(i) * \phi(i, j) \quad (3.2)$$

$$\phi(i, j) = \frac{\Delta(i)}{\sqrt{\Delta^T(i) \Delta(i)}} \quad (3.3)$$

下图为 BFO 算法的趋化操作流程，参数  $m$  是计数器，初始化设  $i = 0$ 。

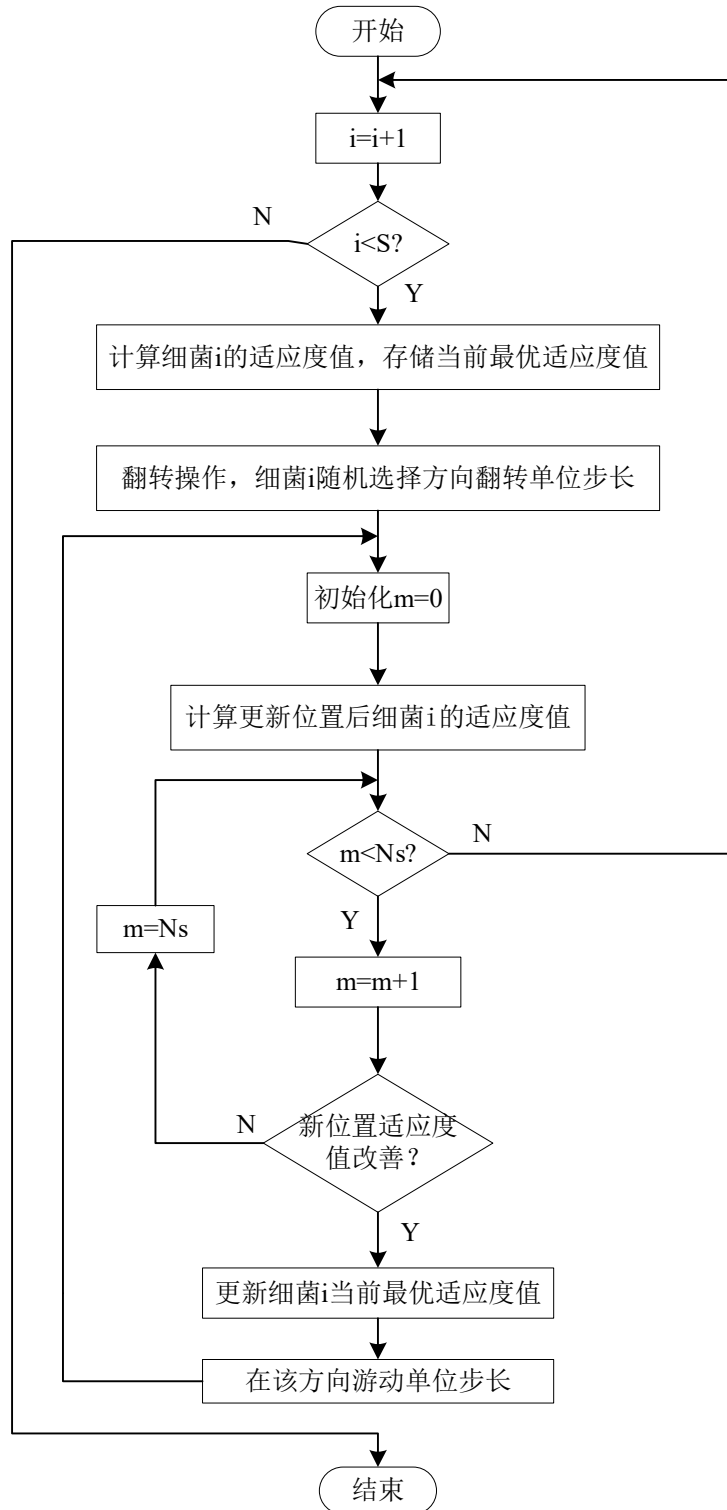


图 3-2 趋化操作流程

### 3.2.2 繁殖行为

BFO 算法执行一段时间的觅食操作后，部分觅食能力弱（适应度值高，本文主要以函数极小值为描述对象）的细菌会被自然淘汰。为维持种群规模不变，剩余的觅食能力强（适应度值低）的细菌会自我繁衍。

对给定的  $k, l$  及每个  $i = 1, 2, \dots, S$ ，定义如下：

$$J_{health}^i = \sum_{j=1}^{N_c+1} J(i, j, k, l) \quad (3.4)$$

式(3.4)为细菌  $i$  的健康度函数，健康度指细菌个体在所有趋化周期内适应度函数值的总和，若其值越小，说明健康度越好。若细菌的健康度差，说明其觅食能力较弱，应被淘汰掉；若健康度好，说明细菌的觅食能力强，将被选择用来繁殖下一代。

设菌群总数为  $S$ ，经典 BFO 中取被淘汰掉的细菌数目  $S_r$  为菌群规模的一半。对所有细菌按照健康度优劣进行排序后，排在前面的一半细菌进行复制，产生与自己具有相同生理特性的后代，而排在后面一半的健康度较差的细菌消亡，由此维持菌群规模不变。

BFO 算法的繁殖操作流程如图如下：

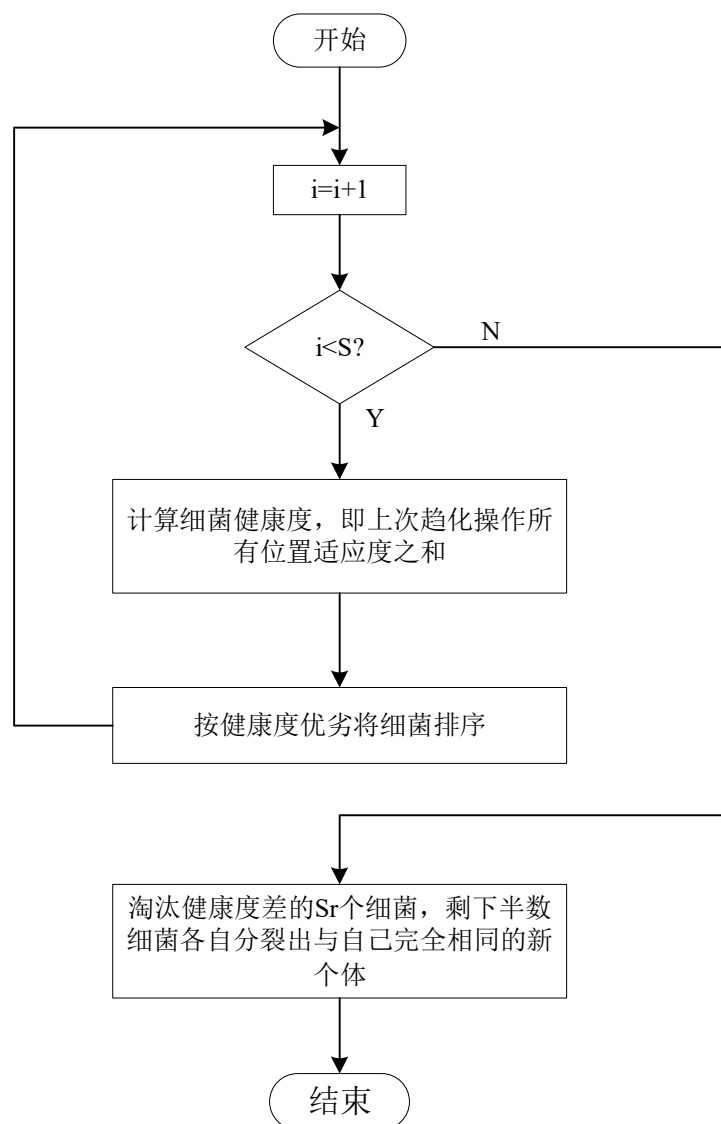


图 3-3 繁殖操作流程

### 3.2.3 迁徙行为

实际环境中的细菌所生活的局部区域可能会发生逐渐变化（如食物耗尽）或者发生温度、酸碱度突然升高等突变，会导致生活在这个局部区域的细菌种群迁徙到新的区域中去，或者集体被外力杀死。

迁徙操作虽会破坏细菌的趋向行为，但细菌也可能因此找到食物更优的区域。故从长远来看，迁徙操作时有益于细菌觅食的。为模拟这一过程，在算法中菌群经过若干代繁殖后，会以给定概率 $P_{ed}$ 执行迁徙操作，被随机重新分配到搜索空间。随机产生的新个体可能更接近全局最优解，由此使得细菌有机会得到更好的适应度函数值，同时解决趋化操作中细菌早熟收敛和陷入局部最优的问题。

如果菌群中的个体满足迁徙条件：即对个体随机生成的概率值小于事先预定好的基准迁徙概率，则该细菌消亡，同时随机地在解空间的任意位置产生一个新个体，以保证菌群总体规模的不变性。



设 $P_{ed}$ 为迁徙概率， $S$ 为菌群规模大小， $rand$ 为 $[0,1]$ 间随机数，BFO 算法的迁徙操作流程如下：

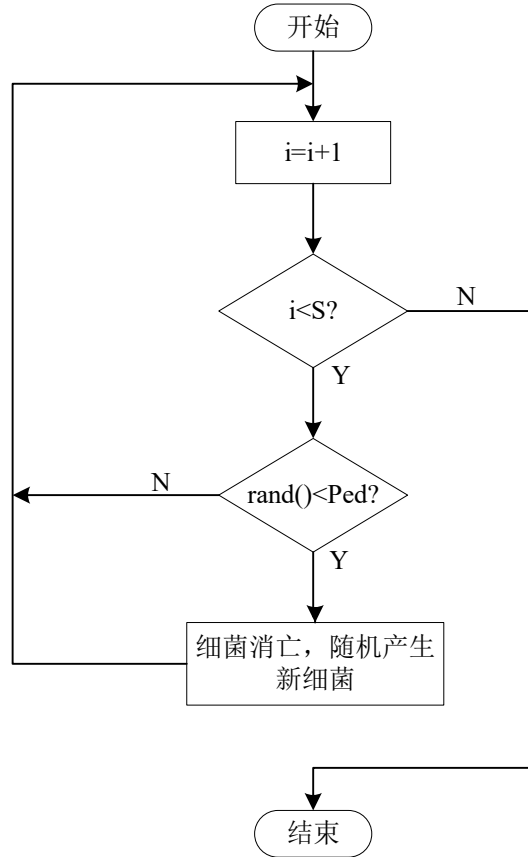


图 3-4 迁徙操作流程图

### 3.3 细菌觅食算法流程

细菌觅食优化算法主要流程如下：

步骤 1：初始化参数 $p$ 、 $S$ 、 $N_c$ 、 $N_s$ 、 $N_{ed}$ 、 $P_{ed}$ 、 $C(i)(i = 1, 2, \dots, S)$ ；

步骤 2：初始化菌群位置，计算每个细菌初始化适应度函数值；

步骤 3：设置循环变量，迁徙循环次数 $l = 1:N_{ed}$ ，繁殖循环次数 $k = 1:N_{re}$ ，趋化循环次数 $j = 1:N_c$ ；

步骤 4：进入趋化循环。

1) 令细菌 $i$ 如下趋向一步， $i = 1, 2, \dots, S$ 。

2) 计算适应度函数值 $J(i, j, k, l)$

3) 令 $J_{best} = J(i, j, k, l)$ ，存储为细菌 $i$ 目前最好的适应值

4) 翻转：生成一个随机向量 $\Delta(i)$ ，翻转方向后继续计算适应度函数值

5) 游动：以公式（3.2）定义向前游动一定步长，继续计算适应度值

步骤 5：进入繁殖循环。将细菌健康度值按增序排列，淘汰掉 $S_r$ （菌群规模一半）个健康度较差的细菌，选取剩下一半健康度较好的细菌进行繁殖，每个细菌分裂成两个完全相同的细菌。

步骤 6: 进入迁徙循环。繁殖周期结束后, 以随机概率 $P_{ed}$ 进行迁徙, 并按(3.1)初始化新的个体。

步骤 7: 达到迁徙次数, 循环结束, 输出结果, 否则转第 4 步, 继续搜索直到所有迭代完成。

算法流程图如下:

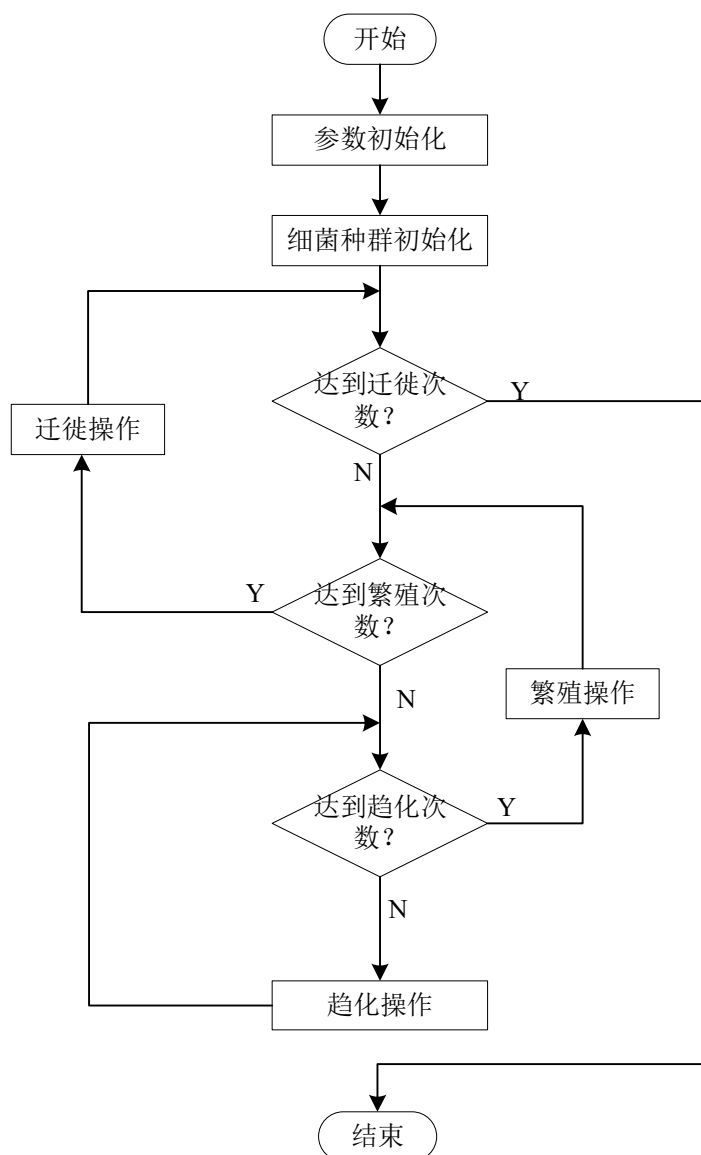


图 3-5 细菌觅食算法总流程图

### 3.4 细菌觅食算法的优缺点

上一小节详细阐述了经典细菌觅食算法的流程, 大量研究和实验以及在工程领域的应用证明了 BFO 算法的可行性和有效性, 其优点在于:

- (1) 局部搜索能力强
- (2) 易于实现和理解
- (3) 具有本质并行性

但是，细究算法的各个操作步骤，影响算法寻优性能的因素在于以下几点：

（1） 菌群初始化

随机布点的方式初始化菌群，无法广泛均匀地覆盖到整个搜索空间，造成菌群多样性不高，降低了算法的全局搜索能力。

（2） 趋化方向选取、趋化步长、游动次数

随机选取趋化方向，盲目性较大，无法体现细菌的认知思考；趋化步长固定，过大影响寻优精度，过小则细菌个体搜索范围小，影响收敛速度；游动次数固定，无论适应度值改善还是变差，游动一定距离即重新搜索，无从经验中学习的能力。

（3） 迁徙概率

固定的迁徙概率，基准概率 $P_{ed}$ 设置过大则细菌频繁迁徙，即“精英逃逸”现象；过小则易陷入局部极值，进化停滞，影响算法收敛能力。

### 3.5 本章小结

本章首先简单介绍了细菌觅食机制的生物学原理，然后研究了 BFO 算法的寻优过程，给出了该算法的详细步骤与流程，最后分析了经典 BFO 中影响算法性能的因素，指出了后续对其进行改进研究工作的方向和必要性。

## 第4章 细菌觅食算法改进研究

针对上一章指出的影响 BFO 算法性能的几个因素及存在的缺陷,本章将从以下几个方面对经典 BFO 进行改进:

- (1) 随机化佳点集方法进行菌群初始化, 保证生成菌群的多样性和均匀性;
- (2) 引入粒子群算法中粒子的个体认知和社会学习机制改进趋化方向选择策略, 引入人工鱼群算法中的聚集操作添加其他细菌对个体的影响, 自适应趋化步长策略改善细菌的经验学习能力;
- (3) 自适应迁徙概率机制, 解决收敛过程中“精英逃逸”现象和进化停滞、陷入局部极值现象。

### 4.1 菌群初始化改进

#### 4.1.1 佳点集法生成菌群

群智能优化算法中的种群初始化操作影响算法在后续的全局搜索能力<sup>[19]</sup>, 随机化生成保证了种群的多样性, 但是无法均匀覆盖到全部搜索空间, 文献[20]中提出的佳点集法可使点集在解空间中均匀分布, 且使点集精度和问题维数无关。

佳点集法初始化菌群的具体步骤如下 (其中,  $l_k$  和  $u_k$  为细菌每一维取值的下限和上限):

- (1): 给定初始种群的大小  $n$ , 以及维数  $s$ , 第 1 个点的第  $k$  维的值为:

$$r_k^1 = e^k - [e^k], \quad 1 < k < s \quad (4.1)$$

构造点  $r = (r_1, r_2, r_3, \dots, r_s)$ , 由佳点集定理可知, 点  $r$  为空间中的佳点。

- (2): 以  $r$  为基础, 第  $i$  个点的第  $k$  维值为:

$$r_k^i = i * r_k - [i * r_k], \quad 1 < k < s, \quad i = 2, 3, \dots, N \quad (4.2)$$

得空间中  $N$  个点佳点集  $P_N = \{r^1, r^2, \dots, r^N\}$ .

- (3): 将佳点集映射到问题的可行域, 则第  $i$  个细菌的第  $k$  维的值为:

$$x_k^i = l_k + r_k^i(u_k - l_k), \quad 1 < k < s, \quad i = 2, 3, \dots, N \quad (4.3)$$

根据以上方法便可得到均匀分布的  $N$  个细菌。

佳点集方法的缺陷在于生成的菌群在搜索空间中线性递增分布, 细菌个数一致则每次得到的结果一致, 无随机性, 无法保证生成菌群的多样性, 搜索空间中的很多点也无法取到。

#### 4.1.2 随机化佳点集法

在佳点集法的基础上, 加上一个随机平移量, 即得到随机化佳点集法, 过程如下:

(1): 设维数为 $d$ , 菌群规模 $S$ , 采用佳点集方法生成 $S$ 个 $d$ 维细菌, 构成初始点集, 每个细菌的取值范围为 $[l_k, u_k]$ , 第 $k$ 个细菌的位置向量 $P_k = [x_k^1, x_k^2, \dots, x_k^d]$ ;

(2): 从多项分布 $\begin{pmatrix} l_k & \dots & u_k \\ \vdots & \ddots & \vdots \\ 1/S & \dots & 1/S \end{pmatrix}$ 中抽取 $d$ 个样本 $\eta_1, \eta_2, \dots, \eta_d$ ;

(3): 令 $P_k = [x_k^1, x_k^2, \dots, x_k^d]$ , 随机平移量 $\Delta$ 为 ( $\{x\}$ 表示取 $x$ 的小数部分):

$$\Delta = \left\{ \frac{x_k^j + \eta_j}{S} \right\}, j = 1, \dots, d \quad (4.4)$$

(4): 对细菌的位置向量进行模 1 随机平移, 得到新位置矢量:

$$P_k = P_k + \Delta \quad (4.5)$$

由此得到的点集 $p_s = P_k | P_k = [x_k^1, x_k^2, \dots, x_k^d], k = 1, \dots, S$ 即经随机化佳点集方法生成的点集。

随机化佳点集法很好地兼顾了覆盖全局搜索空间和布点的随机性两个方面, 使生成的菌群均匀又不失多样性。

## 4.2 趋化行为改进

BFO 算法中趋化行为是算法核心操作, 控制细菌的觅食搜索, 决定了算法的整体寻优能力, 繁殖和迁徙操作起辅助作用, 引导菌群向更好方向发展及防止陷入局部极值<sup>[14]</sup>。

经典 BFO 中的趋化策略是随机选择方向, 固定趋化步长, 使得细菌个体在搜索的时候具有很大的盲目性, 影响算法寻优精度。而固定趋化步长不利于算法的收敛能力。

BFO 算法的趋化行为特性使得其局部搜索能力强, 但是缺乏个体间信息共享与交流机制, 造成全局搜索能力弱。而粒子群算法的优点即在于其个体认知和社会学习机制, 使得其具有良好的全局搜索能力, 所以想到将粒子群算法的核心思想引入到 BFO 算法中, 增强细菌个体的学习能力, 以提高寻优精度。并设计了自适应游动次数策略, 以提高收敛速度。

### 4.2.1 粒子群算法 (PSO)

粒子群算法 (particle swarm optimization, PSO) 模拟鸟类的觅食行为以实现寻优过程<sup>[15][18]</sup>。算法中每个粒子均可记忆自身飞过最优位置 $pbest$ 和全局最优位置 $gbest$ , 粒子飞行受到这两个力的合力的牵引。

设目标搜索空间维度为 $d$ , 粒子 $i$ 的位置为 $x_i = (x_{i1}, x_{i2}, \dots, x_{id})^T$ , 速度为 $v_i = (v_{i1}, v_{i2}, \dots, v_{id})^T$ , 则速度以下式更新:

$$v_{it}^{k+1} = \omega v_{it}^k + c_1 rand_1^k (pbest_{it}^k - x_{it}^k) + c_2 rand_2^k (gbest^k - x_{it}^k) \quad (4.6)$$

$$x_{it}^{k+1} = x_{it}^k + v_{it}^{k+1} \quad (4.7)$$

其中,  $\omega$ 是惯性权重系数,  $t = 1, 2, \dots, k$ ,  $k$ 是迭代次数,  $rand_1, rand_2$ 是 $[0, 1]$ 之间的随机数,  $c_1$ 和 $c_2$ 是加速因子, 用来调节粒子向局部最优  $pbest$  和全局最优  $gbest$  方向飞行的最大步长, 使粒子具有自我总结和向群体中优秀个体学习的能力, 经典 PSO 中建议采用 $c_1 = c_2 = 2$ 。

粒子在 $pbest$ 和 $gbest$ 的制约下的实际飞行轨迹如下图所示：

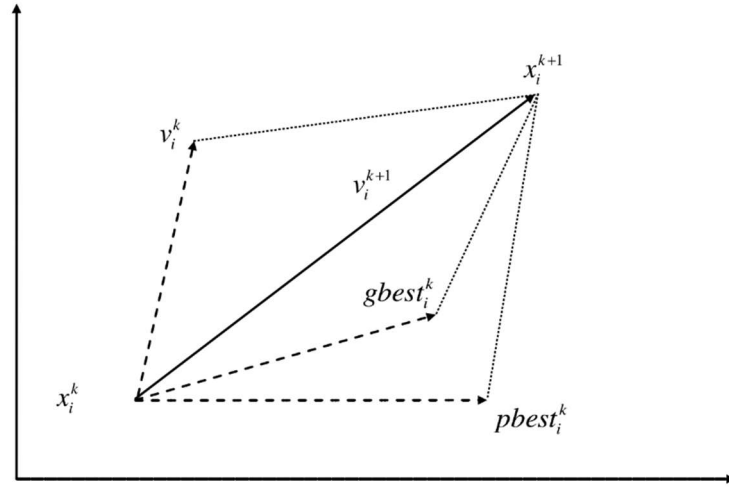


图 4-1 粒子群算法中粒子受合力牵引示意图

#### 4.2.2 趋化方向选择策略改进

本文引入 PSO 算法中的个体认知和社会学习机制以及变权重系数策略，以指导细菌趋化操作中的方向选择。细菌按照下式进行位置更新：

$$\phi(i) = \omega * \phi(i)_{rand} + c_1 * rand_1 * \phi(i)_{pbest} + c_2 * rand_2 * \phi(i)_{gbest} \quad (4.8)$$

$$\phi(i)_{pbest} = \frac{P_i - P_{pbest}}{\sqrt{(P_i - P_{pbest})^T (P_i - P_{pbest})}} \quad (4.9)$$

$$\phi(i)_{gbest} = \frac{P_i - P_{gbest}}{\sqrt{(P_i - P_{gbest})^T (P_i - P_{gbest})}} \quad (4.10)$$

其中， $P_{best}$ 为细菌 $i$ 经历过的个体历史最优位置， $P_{gbest}$ 为菌群经历过的全局历史最优位置， $\phi(i)_{pbest}$ 为个体最优方向矢量， $\phi(i)_{gbest}$ 为全局最优方向矢量， $rand_1, rand_2$ 为 $[0,1]$ 之间随机数， $c_1, c_2$ 为随迭代次数动态改变的学习因子，变化规律为 $[\pi, 2\pi]$ 区间内的余弦函数形式， $\omega$ 为惯性权重系数，作用是权衡全局优化能力和局部优化能力。

设 $N_c$ 为趋化次数， $i$ 为趋化循环的控制变量， $\omega$ 满足：

$$\omega = \sin\left(\frac{\pi}{2} * \frac{N_c - i}{N_c}\right), i = 1, \dots, N_c \quad (4.11)$$

由公式(4.11)可知，趋化操作刚开始进行时， $i = 1, \frac{N_c - i}{N_c} \approx 1$ ，则 $\omega = 1$ ，随着趋化操作的进行， $\omega$ 快速下降，当趋向次数达到最大值 $N_c$ 时 $\omega = 0$ 。

公式(4.8)由三部分组成，第一部分为方向探索部分；第二部分为细菌的个体认知部分，体现了经验学习能力；第三部分为社会学习部分，体现了个体间的信息共享和合作交流机制。

公式(4.8)表示细菌正是根据其随机产生的方向、它自身获得的最好经验以及群体目前的最好经验三方面信息来综合决定下一次趋化操作的方向，然后再按照公式(3.2)更新位置。

对公式(4.8)的几个系数不同取值分情况讨论：

- (1) 公式没有第一部分，即 $\omega = 0, c_1, c_2 \neq 0$ ，细菌只有个体认知和社会学习，只会在一个有限区域进行觅食，而丧失了随机化的全局寻优能力；
- (2) 公式没有第二部分，即 $\omega \neq 0, c_1 = 0, c_2 \neq 0$ ，细菌有全局寻优能力和社会学习能力，但是缺乏个体认知，易出现“盲从”现象；
- (3) 公式没有第三部分，即 $\omega \neq 0, c_1 \neq 0, c_2 = 0$ ，细菌拥有个体认知和全局搜索能力，但是个体间缺乏信息共享和交流，极大降低了寻优精度和收敛速度；
- (4) 公式没有第二、第三部分，即 $c_1 = c_2 = 0$ ，算法退化成经典 BFO 算法。

综上所述，采用公式(4.8)决定的趋化方向较好地平衡了局部寻优能力和全局寻优能力。

- (1) 惯性权重系数 $\omega$ 在趋化循环刚开始时取值最大，下降较缓慢，随着迭代次数的增加迅速下降直至 0；
- (2) 学习因子 $c_1, c_2$ 由最小值经过“慢-快-慢”的斜率变化趋势增长至最大值，符合算法收敛过程的寻优规律：即觅食刚开始时，细菌主要进行随机翻转的“勘探”工作，此时 $\omega$ 下降缓慢， $c_1, c_2$ 上升缓慢，在一定时间内  $\phi(i)_{rand}$ 起主要指导作用；随着趋化次数的增加，菌群对最优解所处区域逐渐有了清晰认知，此时 $\omega$ 快速下降， $c_1, c_2$ 快速上升，公式中起主导作用的部分迅速切换，社会学习及个体认知开始逐渐成为方向的主要考虑因素；
- (3) 最后，当趋化操作即将结束时， $c_1, c_2$ 接近最大值，帮助未收敛至最优解附近的细菌快速向优秀个体靠拢，进一步提高搜索精度。

#### 4.2.3 自适应游动次数策略

固定的游动次数使得细菌在游动过程中，若向着正确的方向游动，适应度值一直改善，也会在一定次数后停止，重新开始选择方向，放弃了正确的道路，增加了不必要的寻优时间，降低搜索效率。

故本文提出自适应游动次数策略：

- (1) 细菌游动一次更新位置，计算适应度值，与历史适应度值比较，若一直改善，则说明游动方向正确，继续游动，直至适应度值不再改善；
- (2) 停止，执行翻转操作，重新按照公式(4.8)选择趋化方向，计算适应度值，改善则游动。

对经典 BFO 算法的趋化操作改进总结如下：

- (1) 将 PSO 算法的思想引入趋化方向选择操作，使其具备个体认知和社会学习、信息共享能力，加快细菌向优秀个体靠拢，又有随机全局寻优能力。
- (2) 同时为了避免菌群陷入局部极值，对趋化方向的惯性系数和学习因子赋以不同权重，使得菌群在刚开始收敛时趋化方向以随机产生为主，随着迭代次数的增加，为帮助菌群快速收敛至最优解，细菌的个体认知和信息共享将对趋化方向起决定性作用。
- (3) 自适应游动次数策略的可以更进一步地加快算法的收敛速度，提高寻优精度。

### 4.3 引入聚集行为

在人工鱼群算法中，鱼群给他通过相互之间的作用来达到聚集行为<sup>[17]</sup>，将其引入BFO算法，施加细菌个体与个体之间的影响，提高细菌智能性。

使细菌与细菌之间既有引力又有斥力，引力使细菌聚在一起，斥力使得每个细菌都有一定的位置，令其能在该位置上获取食物，来维持生存。

细菌间聚集行为用数学公式表达如下：

$$\begin{aligned}
 J_{cc}(\theta, P(j, k, l)) &= \sum_{i=1}^S J_{cc}(\theta, \theta^i(j, k, l)) \\
 &= \sum_{i=1}^S \left[ -d_{\text{attractant}} \exp \left( -w_{\text{attractant}} \sum_{m=1}^p (\theta_m - \theta_i^m)^2 \right) \right] + \\
 &\quad \sum_{i=1}^S \left[ h_{\text{repellant}} \exp \left( -w_{\text{repellant}} \sum_{m=1}^p (\theta_m - \theta_i^m)^2 \right) \right]
 \end{aligned}$$

式中， $d_{\text{attractant}}$ 为引力深度， $w_{\text{attractant}}$ 为引力宽度， $h_{\text{repellant}}$ 为斥力高度， $w_{\text{repellant}}$ 为斥力宽度， $\theta_i^m$ 为细菌 $i$ 的第 $m$ 个分量， $\theta_m$ 为整个菌群中其他细菌的第 $m$ 个分量。

上式的实质是描述了整个菌群在细菌 $i$ 所处位置的作用力之和。

一般情况下，取 $d_{\text{attractant}} = h_{\text{repellant}}$ 。

由于 $J_{cc}(\theta, P(j, k, l))$ 表示种群细菌之间传递信号的影响值，所以在趋向性循环中引入聚集操作后，计算第 $i$ 个细菌的适应度函数值为：

$$J(i, j+1, k, l) = J(i, j, k, l) + J_{cc}(\theta^i(j+1, k, l), P(j+1, k, l)) \quad (4.12)$$

细菌通过上式来修正自身适应度值，以达到聚集目的。

### 4.4 迁徙行为改进

群智能优化算法中往往普遍具有在优化多极值问题时易陷入局部机制的缺陷，BFO算法本身的迁徙机制正是为解决这个问题而设置的。但其考虑不周的地方在于迁徙概率的设定是固定的，其值极大依赖于人工选取。设置过大，可方便菌群跳出局部极值，但是也可能使已经在全局最优位置的个体放弃该位置，即“精英逃逸”现象；设置过小，会导致菌群易陷入进化停滞。

基于以上分析，本节提出自适应迁徙概率策略，同时使用随机化佳点集方法产生新个体。

#### 4.4.1 自适应迁徙概率策略

种群是否陷入早熟收敛可以通过计算两次迭代过程中的群体适应度方差 $\sigma^2$ 来确定，若相邻 $\sigma^2$ 的差值小于一定的阈值 $\text{eps} = 10^{-6}$ ，则可以认为菌群陷入了局部极值，应增加



迁徙概率，对菌群进行扰动；若大于这一阈值，说明菌群寻优效果良好，应减小迁徙概率，不去干扰菌群寻优。

因此，可以让基准概率 $P_{ed}$ 可以根据 $\sigma^2$ 情况自适应调整。设计策略如下：

$P_{ed}$ 的快速增加过程为：

$$P_{ed} = P_{ed} + \Delta \quad (4.13)$$

迅速下降过程为：

$$P_{ed} = \min\{a, \frac{1}{2} \times P_{ed}\} \quad (4.14)$$

$P_{ed}$ 的范围为 $[a, a+b]$ ， $a$  为基准概率， $b$  控制变化增量 $\Delta$ 的上限，即 $\Delta$ 的范围是 $[0, b]$ ， $\Delta$ 遵循如下变化过程：

$$\Delta = b \times (-\exp(-0.1 * i) + 1)$$

其中， $i$  为迁徙循环的迭代控制变量。

(1) 当检测到菌群陷入早熟收敛时， $P_{ed}$ 按照公式(4.13)快速增加，此时  $\Delta$ 曲线斜率较大，可保证 $P_{ed}$ 值的快速变化，以促进菌群尽快跳出局部极值；

(2) 当 $P_{ed}$ 增大至接近上限时，曲线斜率趋于平缓， $P_{ed}$ 增长幅度减慢，符合变化规律。

(3) 迁徙过程中，当检测到菌群相邻两次的群体适应度方差大于  $eps$  时，即可判断菌群已跳出早熟收敛， $P_{ed}$ 按照(4.14)迅速减小。

重复这样的自适应变化过程直到达到最大迁徙次数为止。

#### 4.4.2 改进后的迁徙操作

改进后迁徙操作流程图如下：

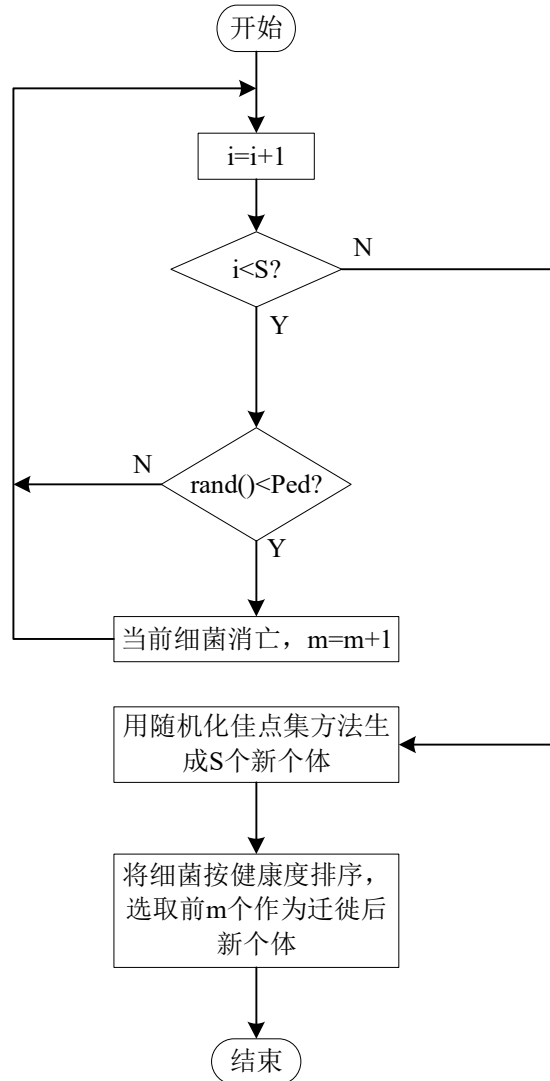


图 4-2 改进后的迁徙操作流程圖

#### 4.5 算法设计与实现

总结前文提出的改进思路，得出改进后的细菌觅食算法流程如下（其中  $i$ 、 $j$ 、 $k$  的初值均为 0，设第  $k$  个细菌的位置向量为  $P_k = [x_k^1, x_k^2, \dots, x_k^d]$ ，每个细菌取值范围为  $[l_k, u_k]$ ）：

步骤 1：初始化参数。细菌规模大小  $S$ 、趋化步长  $C$ 、搜索空间的维数  $d$ 、趋化循环次数  $N_c$ 、复制次数  $N_{re}$ 、迁徙次数  $N_{ed}$ 、基准概率值  $a = 0.1$ ，上限控制变量  $b = 0.3$ 。

步骤 2：菌群初始化。采用随机化佳点集方法初始菌群，并计算每个细菌的适应度值  $f$ 。

步骤 3：设置循环变量。其中迁移循环次数  $l = 1:N_{ed}$ ，繁殖循环次数  $k = 1:N_{re}$ ，趋化循环次数  $j = 1:N_c$ 。

步骤 4：执行趋化操作。

（1）翻转：按照改进后趋化方向选择策略选取方向，并翻转，计算适应度值；

(2) 游动：执行自适应游动次数策略进行游动。

步骤 5：执行繁殖操作。计算健康度并排序，健康度小的一半群体繁殖。

步骤 6：执行迁徙操作。按自适应迁徙概率策略进行迁徙，并以随机化佳点集方法生成新个体。

步骤 7：算法结束判断操作。

综上，改进后的算法的流程图如下：

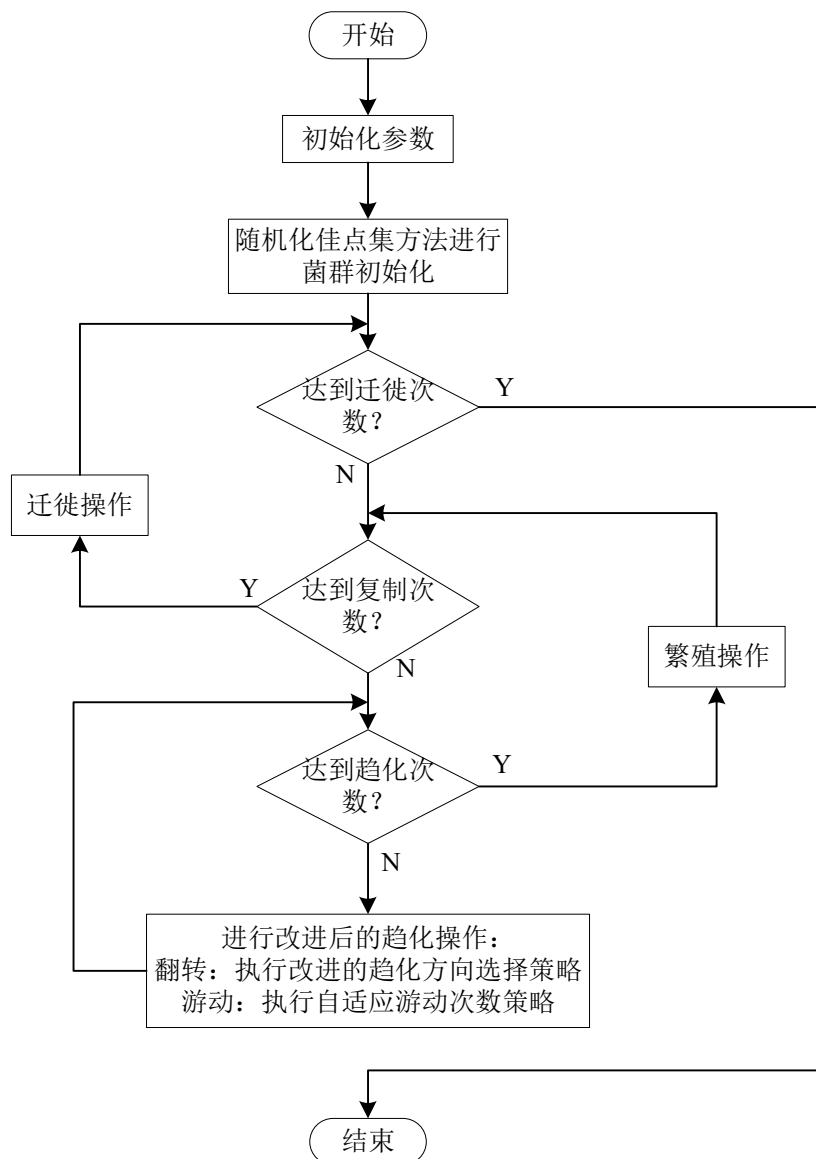


图 4-3 改进算法总流程图

## 4.6 本章小结

本章针对经典 BFO 的一些缺陷，提出了四个方面的改进措施：

(1) 随机化佳点集方法生成初始菌群，保证菌群分布的均匀性、多样性及全局寻优性。

（2）粒子群算法思想改进趋化方向选择策略以及自适应游动次数策略提高算法的求解精度和收敛速度；

（3）引入了聚集操作，添加其他细菌对个体的影响；

（4）自适应迁徙概率策略能够及时检测到菌群是否陷入早熟收敛，帮助算法更好地寻优，同时避免了精英细菌的逃逸现象，加速算法收敛过程。

以上述改进思路设计了改进后算法，并给出了算法步骤和流程图。并设计了实验证明了改进算法在性能上的优越性。

## 第5章 改进算法应用于灰度图像增强问题

本章将改进后的细菌觅食算法应用于解决灰度图像空域增强问题中，以增加空域方法的自适应能力，得到最优的增强效果。

### 5.1 灰度图像增强问题

#### 5.1.1 基于变换的灰度图像增强问题

传统的空域增强方法，无论是线性变换、分段线性变换，还是对数变换、伽玛变换，其变换函数需要人为设置，且不能完美拟合所有类型的图像，缺乏自适应能力，应用受限。

Tubbs 提出的不完全 beta 函数可以很好地解决这个问题，增加了空间域方法灰度变换函数选取的灵活性。不完全 beta 函数中对两个参数取不同的值，对应下面四种灰度变换曲线：

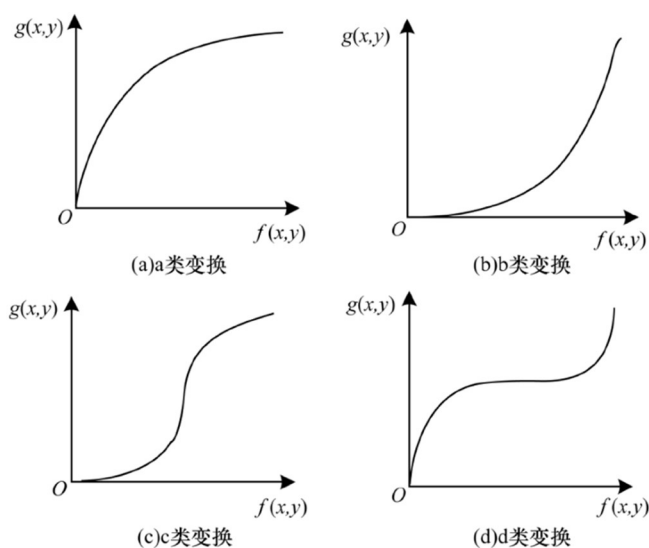


图 5-1 四种灰度变换曲线

- (a) 对图像较暗区域进行拉伸
- (b) 对图像较亮区域进行拉伸
- (c) 中间拉伸两端压缩
- (d) 两端拉伸中间压缩

用不完全 beta 函数拟合图像变换曲线需要先对图像进行归一化操作<sup>[22]</sup>，归一化的变换函数  $F(u)$  定义为：

$$F(u) = B^{-1}(\alpha, \beta) \int_0^u t^{\alpha-1} (1-t)^{\beta-1} dt, 0 < \alpha, \beta < 10 \quad (5.1)$$

其中， $u$  为原始图像像素， $B(\alpha, \beta)$  为 beta 函数：

$$B(\alpha, \beta) = \int_0^1 t^{\alpha-1} (1-t)^{\beta-1} dt \quad (5.2)$$

确定不完全 beta 函数两个参数的过程并不直观，没有统一的方法。传统方法包括穷举法和人工介入法，但该类方法无智能性、无自动化性能，计算量大、效率低，且人工设置参数的正确度无法保证、直观性太强。

因此，相关学者提出了利用群智能优化机制自动拟合不完全 beta 函数的思路<sup>[7]</sup>。本文利用 BFO 算法较强的局部搜索能力实现参数  $\alpha, \beta$  的动态确定，并将不完全 beta 函数的参数确定问题转化为最优化问题。

## 5.2 改进算法在灰度图像增强问题上的应用

### 5.2.1 适应度函数选择

适应度函数决定优化算法中个体寻优进化的动力，常用的图像增强度量考虑以下几个重要因素：信息熵、紧致度、信噪改变量以及像素差别，其各自定义如下：

- (1) 均方误差  $F_{ac}$
- (2) 信息熵  $E$
- (3) 像素差别  $F_{br}$
- (4) 信噪改变量  $I_{nc}$
- (5) 紧致度  $C$

本文根据文献[3]的建议，设计能够更全面反映图像信息的适应度函数如下：

$$f_{Fitness} = E \times I_{nc} \times [F_{ac} + 2.5 \times C] + F_{br} \quad (5.3)$$

$f_{Fitness}$  兼顾了图像五个方面的度量标准，实验证明，采用该适应度函数与传统只使用均方误差作为适应度函数相比，能得到更好的图像增强效果。 $f_{Fitness}$  值越大，说明图像增强后的效果越好。这里对  $f_{Fitness}$  取反，以适应 BFO 算法。

### 5.2.2 采用改进 BFO 的图像增强方法

若采用传统的群智能算法优化目标函数的机制，直接对图像进行优化的话，图像维数一般在百维以上，显然是不合适的，所以可以转换思路，去优化不完全 beta 函数的两个参数，这样细菌维数就固定在 2 维，使得群智能算法进行图像增强成为可能。

设原图像中点  $(x, y)$  的灰度值为  $f(x, y)$ ，增强后灰度值为  $f'(x, y)$ ，则利用本文改进算法实现图像自适应增强的过程如下：

步骤 1：归一化处理：

$$u(x, y) = \frac{f(x, y) - i_{min}}{i_{max} - i_{min}} \quad (5.4)$$

步骤 2：对归一化之后的原始图像调用改进 BFO 算法：

- (1) 细菌位置改变，则参数值  $\alpha, \beta$  变化，重新对原图像进行拟合：

$$u'(x, y) = F(u(x, y)) \quad (5.5)$$

其中， $0 \leq u'(x, y) \leq 1$ ，再按下式反归一化操作，变换至  $[0, 255]$  区间：

$$f'(x,y) = 255 \times u'(x,y) \quad (5.6)$$

(2) 比较 $f'$ 与 $f$ 的适应度值大小，继续算法操作。

步骤 3：算法结束，返回变换后最优适应度值对应图像并输出最佳适应度值和对应的最佳 $\alpha, \beta$ 参数组合。

基于改进 BFO 的图像增强算法流程图如下：

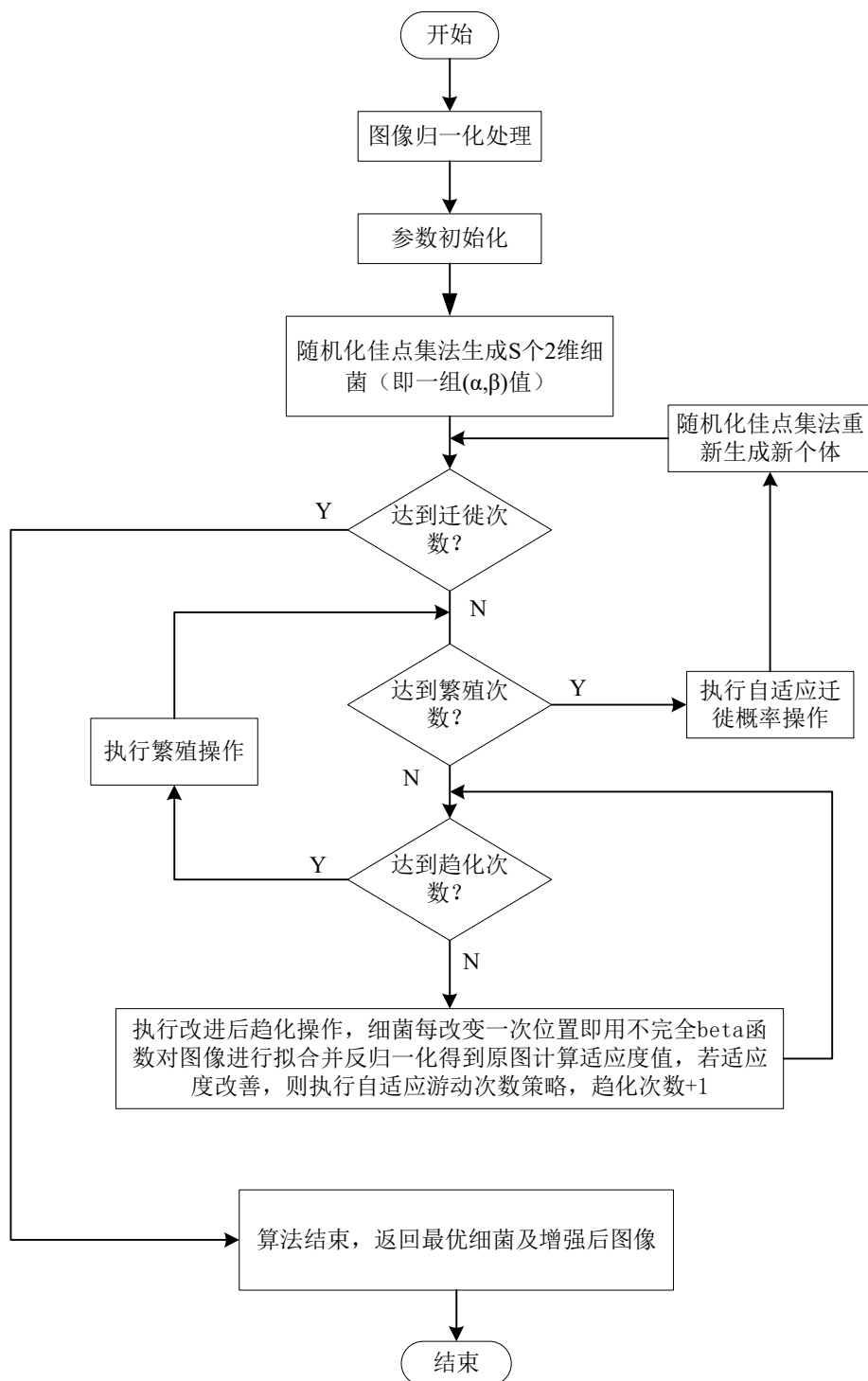


图 5-2 改进 BFO 的图像增强算法流程图

### 5.3 仿真结果与分析

为了验证本文思路的有效性和改进 BFO 算法进行图像增强处理的效果和性能,设计多组实验:

- (1) 分别利用 BFO 及改进 BFO 算法对两幅不同降质类型的灰度图像 **lena** 和 **cameraman** 进行图像增强测试,比较经算法优化得到的 $\alpha, \beta$ 值,对比增强后图像的适应度值优劣、优化速度(进化代数)以及图像在细节处理、明暗分配方面的质量;
- (2) 将采用改进 BFO 算法的优化结果直方图与采用传统直方图法、对数变换、伽玛变换,以及文献中提到的算法优化结果进行对比分析,验证增强后图像的效果。

#### 5.3.1 算法参数设置

算法参数设置如下:

维度 $d = 2$ , 菌群数 $S = 40$ , 趋化次数 $N_c = 10$ , 繁殖次数 $N_{re} = 5$ , 迁徙次数 $N_{ed} = 5$ , 基准概率 $P_{ed} = 0.1$ , 上限控制变量设置 $b = 0.3$ 。

分别用经典 BFO 算法和改进 BFO 算法对两幅灰度图 **lena** 和 **cameraman** 独立运行 50 次, 并对测试结果进行对比分析。

#### 5.3.2 对比分析结果

细菌菌群在觅食过程中的位置更新示意图如下:



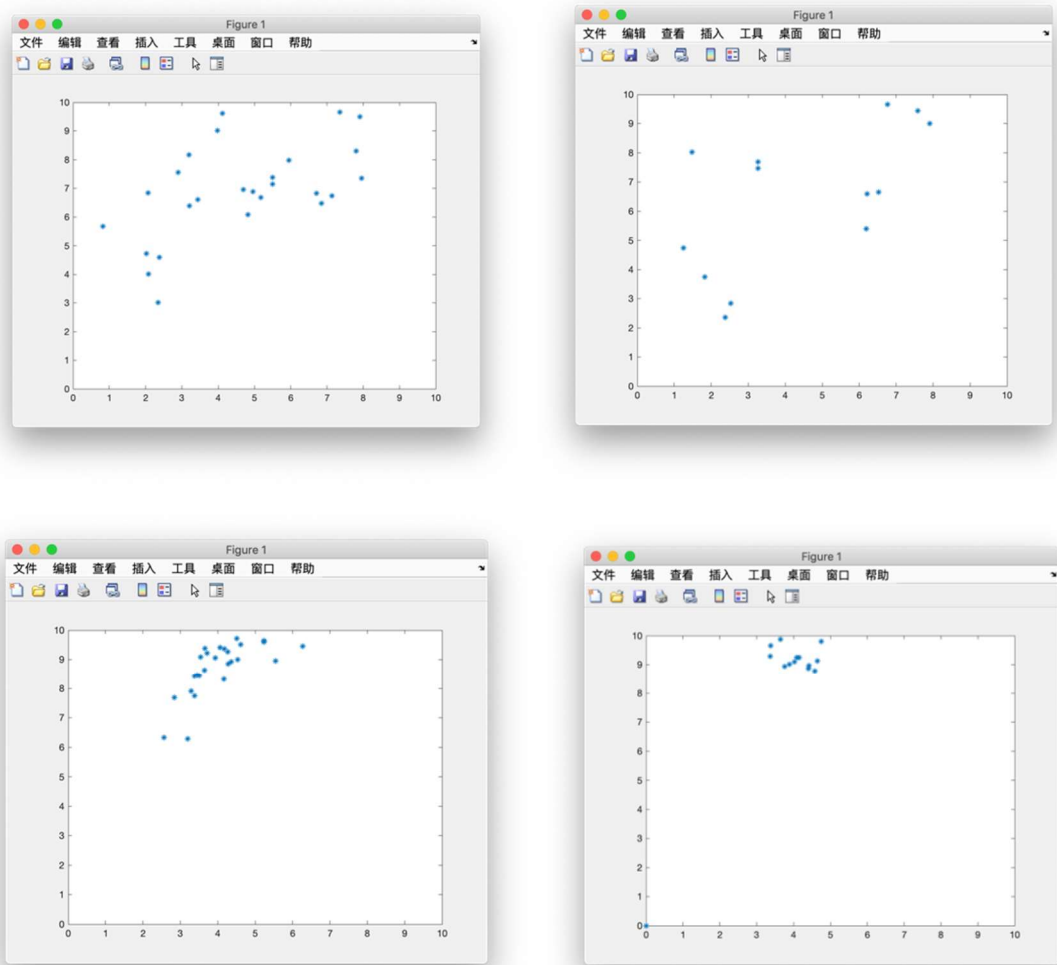


图 5-3 细菌觅食过程中位置更新示意图

程序运行结束，得到增强后效果如图：

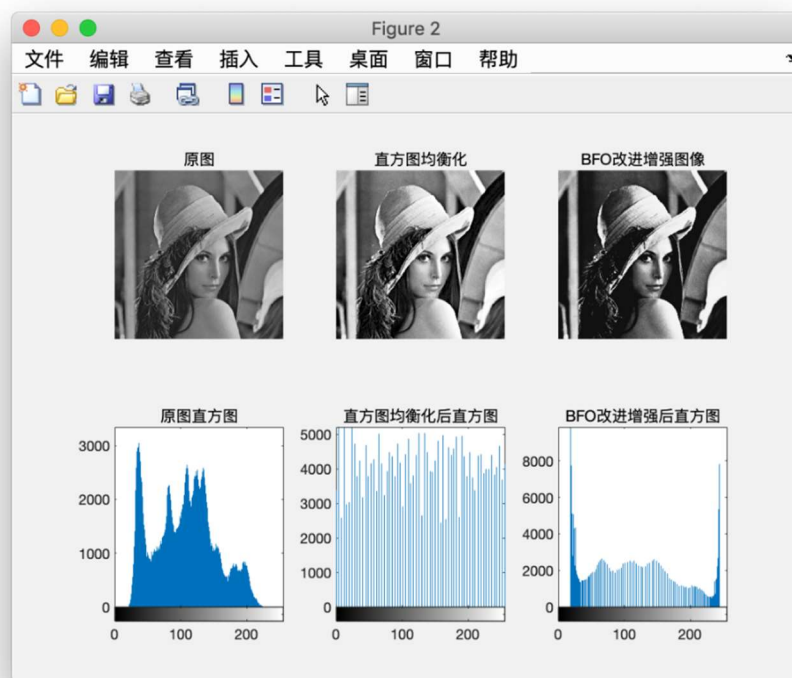


图 5-4 改进 BFO 增强效果对比图（lena）

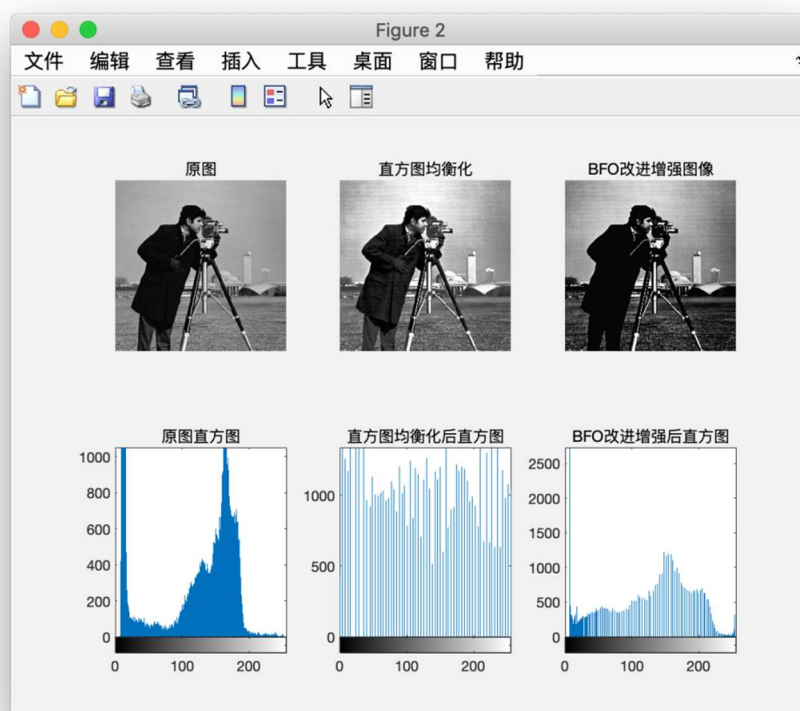


图 5-5 改进 BFO 图像增强效果对比图（cameraman）

#### 5.4 本章小结

本章提出了将改进 BFO 算法优化不完全  $\beta$  函数的两个参数并对灰度变换曲线进行自适应拟合从而得到增强图像的思路，使得高维图像优化问题转换为固定二维参数寻优问题，并设计实验取得了良好的增强效果，验证了本文提出方法的可行性和有效性。

## 结论与展望

本文研究了传统的图像增强方法，针对空域增强方法缺乏自适应性的缺点，将群智能算法中的细菌觅食算法应用于灰度变换增强问题，并研究分析了经典 BFO 算法的原理优点及缺陷，对其存在的不足之处进行了改进，设计了仿真实验验证了本文方法的有效性。

本研究课题的成果如下：

- （1）结合粒子群思想的混合细菌觅食算法
  - 1）随机化佳点集方法构造初始菌群，使得细菌布点均匀又不失多样性
  - 2）结合粒子群算法全局寻优能力强的优点，将个体认知和社会学习思想与细菌趋化行为相结合，并设计了自适应游动次数策略，大大提高了算法的寻优精度和收敛速度。
  - 3）设计了自适应迁徙概率策略，提高了算法的早熟抑制能力，有效避免了“精英逃逸”现象和进化停滞现象，增强了算法的收敛性。
- （2）改进 BFO 算法在图像增强上的应用
  - 1）研究分析了传统图像空域增强方法的特性和不足之处，并做了大量仿真实验。
  - 2）使用不完全 beta 函数去拟合灰度图像增强的变换曲线，将改进后的细菌觅食算法用于不完全 beta 函数参数寻优，在图像细节增强方面取得了良好的增强效果。

细菌觅食算法因提出较晚，研究较少，目前还存在很大的改进空间，受限于课题的时间和本人水平，只做了有限改进，在如下方向仍有进一步探索和研究的必要：

- （1）参数选取问题。群智能算法的参数选取对算法性能影响至关重要，目前多是靠大量重复实验和个人经验，不具有普遍适用性，仍需发掘其中参数设置的规律。
- （2）算法理论的研究和证明。群智能算法的启发性特点，使其虽然在实际工程应用问题上往往能取得令人惊艳的效果，但是其收敛性的证明还没有一个确切可靠的数学理论支撑。目前学术界的研究也取得了一些突破性进展，对其继续研究进行科学合理的证明也是一个值得探索的问题。
- （3）工程领域的应用问题。细菌觅食算法有较强的局部搜索能力和很好的稳定性，在工程领域的应用还不多，在机器学习、电力系统调度、函数优化、图像处理、机器视觉、量化投资等领域问题上的研究，也是一片广阔天地。

## 致 谢

伴随着毕业论文的撰写完成，我也即将要结束这美好的学生时代。回首四年的学习生涯，对那些曾经给予我引导、鼓励和帮助的人，心中充满了感激。

首先我要感谢我的导师窦易文副教授，在这几个月来，正是在窦老师的不断鼓励与指引下，才基本开发完成了算法的改进测试调优工作。在跟着窦老师做毕业设计的几个月中，窦老师扎实的专业功底，严谨的治学态度，精益求精的工作作风令我受益匪浅。从论文的选题到论文的完成，窦老师给与了热心专业的指导，从而我顺利的完成了论文的研究工作，感谢老师对我研究学习上的谆谆教诲，在此郑重地向导师表示我最衷心的感谢和感谢！

感谢我的母校——安徽工程大学给我提供了一个良好的学习环境。母校丰富的资源和优良的学术氛围给了我极大的帮助。

还要感谢我的父母，他们总是毫无保留的给予我支持和鼓励，我的每点进步都是和他们的支持分不开的，他们的殷切期望永远是激励我向上的强大动力和精神支柱。他们的挂念时刻提醒着我，我不是一个人。

感谢同寝室的室友，感谢一直关心与支持我的同学和朋友们，四年来朝夕相处的计算机科学与技术专业的全体同学以及寝室舍友们，感谢你们在学习和工作中给与我的鼓励、帮助和支持，给我的大学生活留下了美好的青春的回忆。

最后，向参加评审的各位专家教授表示真挚谢意，老师们辛苦了！

作者：

年 月 日

## 参考文献

- [1]胡洁. 细菌觅食优化算法的改进及应用研究[D].武汉理工大学,2012.
- [2]翟自勇,赵卫国,王欢.基于自适应细菌觅食算法的灰度图像增强方法[J].河北工程大学学报(自然科学版),2013,30(01):77-81.
- [3]施泽波.图像增强中优化算法适应度函数设计[J].电光与控制,2013,20(05):49-52.
- [4]刘小龙,李荣钧,杨萍.基于高斯分布估计的细菌觅食优化算法[J].控制与决策,2011,26(08):1233-1238.
- [5]林诗洁,董晨,陈明志,张凡,陈景辉.新型群智能优化算法综述[J].计算机工程与应用,2018,54(12):1-9.
- [6]刘小龙. 细菌觅食优化算法的改进及应用[D].华南理工大学,2011.
- [7]孙勇强. 基于群体智能优化算法的图像增强研究[D].江南大学,2008.
- [8]刘珍,孙京浩.一种改进的细菌觅食优化算法[J].华东理工大学学报(自然科学版),2016,42(02):225-232.
- [9]周佳薇. 细菌觅食优化算法研究及其在图像增强中的应用[D].西安电子科技大学,2014.
- [10]群智能优化算法理论与应用[M]. 科学出版社 , 梁艳春, 2009
- [11]数字图像处理[M]. 机械工业出版社 , (美)WilliamK.Pratt 著, 2005
- [12]姜建国,周佳薇,周润生,王娟.一种采用改进细菌觅食优化算法的图像增强方法[J].控制与决策,2015,30(03):461-466.
- [13]谢芬. 基于细菌觅食算法和多目标优化的图像分割与增强[D].南京师范大学,2015.
- [14]姜建国,周佳薇,郑迎春,王涛.一种自适应细菌觅食优化算法[J].西安电子科技大学学报,2015,42(01):75-81.
- [15]梁樱馨,田浩杉.基于细菌觅食与粒子群的改进混合算法[J].电子科技,2017,30(04):79-82.
- [16]张铮,徐超,任淑霞,韩海玲.数字图像处理与机器视觉: Visual C++与 Matlab 实现[M].人民邮电出版社.2014
- [17]刘慧敏.人工鱼群算法研究及其在图像增强中的应用[D].西安电子科技大学,2014.
- [18]孙晶晶. 粒子群优化算法的改进及其应用研究[D].陕西师范大学,2010.
- [19]郑迎春. 细菌觅食优化算法研究[D].西安电子科技大学,2013.
- [20]张铃,张钺.佳点集遗传算法[J].计算机学报,2001(09):917-922.
- [21]晏晓辉,朱云龙,张智聪,吕赐兴,李帅,蚁文洁.菌群优化方法及其应用研究综述[J].郑州大学学报(工学版),2018,39(05):1-10+21.
- [22]郭文艳,周吉瑞,张姣姣.基于改进人工蜂群的图像增强算法[J].计算机工程,2017,43(11):261-271.
- [23]贾占朝. 遗传微粒群混合算法的研究及其在图像增强中的应用[D].中南大学,2009.
- [24]王鑫,孔航,崔雨航,赵春玲,赵大众.基于 Matlab 的图像增强算法研究[J].电脑与信息技术,2018,26(02):31-36.
- [25]Dong Hwa Kim,Ajith Abraham,Jae Hoon Cho. A hybrid genetic algorithm and bacterial foraging approach for global optimization[J]. Information Sciences,2007,177(18).

- [26]Hanning Chen, Yunlong Zhu, Kunyuan Hu. Self-adaptation in bacterial foraging optimization algorithm [C]// Proceedings of the 3rd International Conference on Intelligent System and Knowledge Engineering, 2008, 1: 1026-1031.
- [27]Jun Sun,Wei Fang,Vasile Palade,Xiaojun Wu,Wenbo Xu. Quantum-behaved particle swarm optimization with Gaussian distributed local attractor point[J]. Applied Mathematics and Computation,2011,218(7).
- [28]Mishra, S. A hybrid least square-fuzzy bacterial foraging strategy for harmonic estimation [J]. IEEE Transaction of Evolutionary Computation, 2005, 9(1):61-73.
- [29]Tripathy M, Mishra S. Bacteria Foraging-Based Solution to Optimize Both Real Power Loss and Voltage Stability Limit [J]. IEEE Transactions on Power Systems, 2007, 22(1): 240-248.

## 附录 A 外文文献及翻译

### **Hybrid Genetic Algorithm Based Image Enhancement Technology**

**Abstract**—in image enhancement, Tubbs proposed a normalized incomplete Beta function to represent several kinds of commonly used non-linear transform functions to do the research on image enhancement. But how to define the coefficients of the Beta function is still a problem. We proposed a Hybrid Genetic Algorithm which combines the Differential Evolution to the Genetic Algorithm in the image enhancement process and utilize the quickly searching ability of the algorithm to carry out the adaptive mutation and searches. Finally we use the Simulation experiment to prove the effectiveness of the method.

**Keywords**- Image enhancement; Hybrid Genetic Algorithm; adaptive enhancement

#### I. INTRODUCTION

In the image formation, transfer or conversion process, due to other objective factors such as system noise, inadequate or excessive exposure, relative motion and so the impact will get the image often a difference between the original image (referred to as degraded or degraded) Degraded image is usually blurred or after the extraction of information through the machine to reduce or even wrong, it must take some measures for its improvement.

Image enhancement technology is proposed in this sense, and the purpose is to improve the image quality. Fuzzy Image Enhancement situation according to the image using a variety of special technical highlights some of the information in the image, reduce or eliminate the irrelevant information, to emphasize the image of the whole or the purpose of local features. Image enhancement method is still no unified theory, image enhancement techniques can be divided into three categories: point operations, and spatial frequency enhancement methods Enhancement Act. This paper presents an automatic adjustment according to the image characteristics of adaptive image enhancement method that called hybrid genetic algorithm. It combines the differential evolution algorithm of adaptive search capabilities, automatically determines the transformation function of the parameter values in order to achieve adaptive image enhancement.

#### II. IMAGE ENHANCEMENT TECHNOLOGY

Image enhancement refers to some features of the image, such as contour, contrast, emphasis or highlight edges, etc., in order to facilitate detection or further analysis and



processing. Enhancements will not increase the information in the image data, but will choose the appropriate features of the expansion of dynamic range, making these features more easily detected or identified, for the detection and treatment follow-up analysis and lay a good foundation.

Image enhancement method consists of point operations, spatial filtering, and frequency domain filtering categories. Point operations, including contrast stretching, histogram modeling, and limiting noise and image subtraction techniques. Spatial filter including low-pass filtering, median filtering, high pass filter (image sharpening). Frequency filter including homomorphism filtering, multi-scale multi-resolution image enhancement applied [1].

### III. DIFFERENTIAL EVOLUTION ALGORITHM

Differential Evolution (DE) was first proposed by Price and Storn, and with other evolutionary algorithms are compared, DE algorithm has a strong spatial search capability, and easy to implement, easy to understand. DE algorithm is a novel search algorithm, it is first in the search space randomly generates the initial population and then calculate the difference between any two members of the vector, and the difference is added to the third member of the vector, by which Method to form a new individual. If you find that the fitness of new individual members better than the original, then replace the original with the formation of individual self.

The operation of DE is the same as genetic algorithm, and it conclude mutation, crossover and selection, but the methods are different. We suppose that the group size is P, the vector dimension is D, and we can express the object vector as (1):

$$x_i = [x_{i1}, x_{i2}, \dots, x_{iD}] \quad (i=1, \dots, P) \quad (1)$$

And the mutation vector can be expressed as (2):

$$V_i = X_{r1} + F \times (X_{r2} - X_{r3}) \quad i=1, \dots, P \quad (2)$$

$X_{r1}, X_{r2}, X_{r3}$  are three randomly selected individuals from group, and  $r1 \neq r2 \neq r3 \neq i$ . F is a range of [0, 2] between the actual type constant factor difference vector is used to control the influence, commonly referred to as scaling factor. Clearly the difference between the vector and the smaller the disturbance also smaller, which means that if groups close to the optimum value, the disturbance will be automatically reduced.

DE algorithm selection operation is a "greedy " selection mode, if and only if the new vector  $u_i$  the fitness of the individual than the target vector is better when the individual  $x_i$ ,  $u_i$  will be retained to the next group. Otherwise, the target vector  $x_i$  individuals remain in the original group, once again as the next generation of the parent vector.

### IV. HYBRID GA FOR IMAGE ENHANCEMENT IMAGE

enhancement is the foundation to get the fast object detection, so it is necessary to find real-time and good performance algorithm. For the practical requirements of different systems, many algorithms need to determine the parameters and artificial thresholds. Can use a non-complete Beta function, it can completely cover the typical image enhancement transform type, but to determine the Beta function parameters are still many problems to be solved. This section presents a Beta function, since according to the applicable method for image enhancement, adaptive Hybrid genetic algorithm search capabilities, automatically determines the transformation function of the parameter values in order to achieve adaptive image enhancement.

The purpose of image enhancement is to improve image quality, which are more prominent features of the specified restore the degraded image details and so on. In the degraded image in a common feature is the contrast lower side usually presents bright, dim or gray concentrated. Low-contrast degraded image can be stretched to achieve a dynamic histogram enhancement, such as gray level change. We use  $I_{xy}$  to illustrate the gray level of point  $(x, y)$  which can be expressed by (3).

$$I_{xy}=f(x, y) \quad (3)$$

where: “f” is a linear or nonlinear function. In general, gray image have four nonlinear translations [6] [7] that can be shown as Figure 1. We use a normalized incomplete Beta function to automatically fit the 4 categories of image enhancement transformation curve. It defines in (4):

$$f(u)=B^{-1}(\alpha, \beta) \int_0^u t^{\alpha-1} (1-t)^{\beta-1} dt (\alpha < 0, \beta < 10) \quad (4)$$

where:

$$B(\alpha, \beta) = \int_0^1 t^{\alpha-1} (1-t)^{\beta-1} dt \quad (5)$$

For different value of  $\alpha$  and  $\beta$ , we can get response curve from (4) and (5).

The hybrid GA can make use of the previous section adaptive differential evolution algorithm to search for the best function to determine a value of Beta, and then each pixel grayscale values into the Beta function, the corresponding transformation of Figure 1, resulting in ideal image enhancement. The detail description is follows:

Assuming the original image pixel  $(x, y)$  of the pixel gray level by the formula (4), denoted by  $i_{xy}$ ,  $(x, y) \in \Omega$ , here  $\Omega$  is the image domain. Enhanced image is denoted by  $I_{xy}$ . Firstly, the image gray value normalized into  $[0, 1]$  by (6).

$$g_{xy} = \frac{i_{xy} - i_{\min}}{i_{\max} - i_{\min}} \quad (6)$$

where:  $i_{\max}$  and  $i_{\min}$  express the maximum and minimum of image gray relatively.

Define the nonlinear transformation function  $f(u)$  ( $0 \leq u \leq 1$ ) to transform source image to  $G_{xy} = f(g_{xy})$ , where the  $0 \leq G_{xy} \leq 1$ .

Finally, we use the hybrid genetic algorithm to determine the appropriate Beta function  $f(u)$  the optimal parameters  $\alpha$  and  $\beta$ . Will enhance the image  $G_{xy}$  transformed antinormalized.

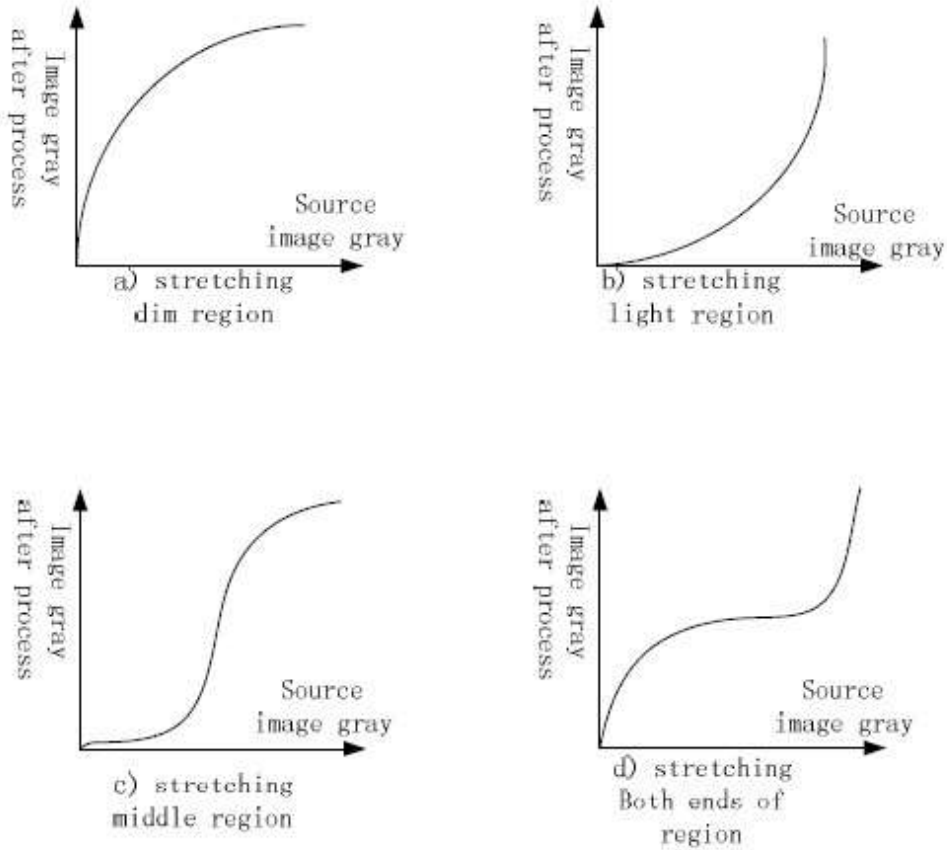


Figure 1. Four traditional translations

## V. EXPERIMENT AND ANALYSIS

In the simulation, we used two different types of gray-scale images degraded; the program performed 50 times, population sizes of 30, evolved 600 times. The results show that the proposed method can very effectively enhance the different types of degraded image.

Figure 2, the size of the original image a  $320 \times 320$ , it's the contrast to low, and some details of the more obscure, in particular, scarves and other details of the texture is not obvious, visual effects, poor, using the method proposed in this section, to overcome the above some of the issues and get satisfactory image results, as shown in Figure 5 (b) shows, the visual effects have been well improved. From the histogram view, the scope of the distribution of image intensity is more uniform, and the distribution of light and dark gray

area is more reasonable. Hybrid genetic algorithm to automatically identify the nonlinear transformation of the function curve, and the values obtained before 9.837,5.7912, from the curve can be drawn, it is consistent with Figure 3, c-class, that stretch across the middle region compression transform the region, which were consistent with the histogram, the overall original image low contrast, compression at both ends of the middle region stretching region is consistent with human visual sense, enhanced the effect of significantly improved.

Figure 3, the size of the original image a  $320 \times 256$ , the overall intensity is low, the use of the method proposed in this section are the images b, we can see the ground, chairs and clothes and other details of the resolution and contrast than the original image has Improved significantly, the original image gray distribution concentrated in the lower region, and the enhanced image of the gray uniform, gray before and after transformation and nonlinear transformation of basic graph 3 (a) the same class, namely, the image Dim region stretching, and the values were 5.9409,9.5704, nonlinear transformation of images degraded type inference is correct, the enhanced visual effect and good robustness enhancement.

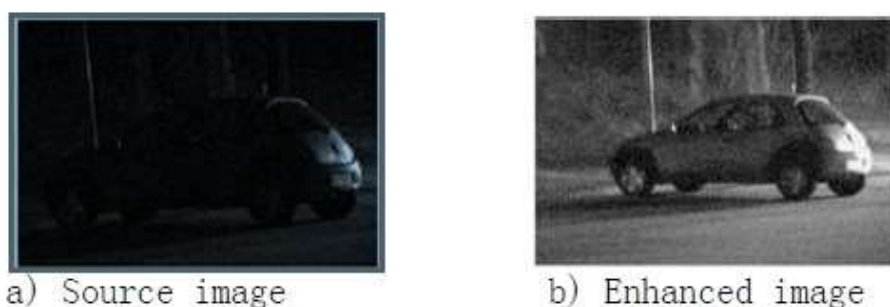


Figure 2. The process of single object

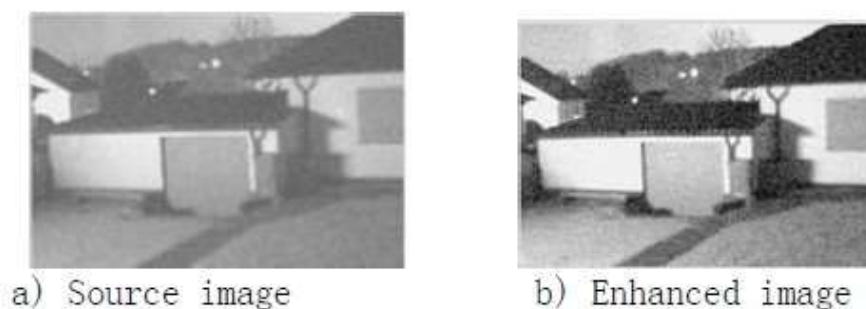


Figure 3. The process of moving objects

Difficult to assess the quality of image enhancement, image is still no common evaluation criteria, common peak signal to noise ratio (PSNR) evaluation in terms of line, but the peak signal to noise ratio does not reflect the human visual system error. Therefore, we use marginal protection index and contrast increase index to evaluate the experimental results.

Edgel Protection Index (EPI) is defined as follows:

$$EPI = \frac{\sum (|I_D(i,j) - I_D(i+1,j)| + |I_D(i,j) - I_D(i,j+1)|)}{\sum (|I_O(i,j) - I_O(i+1,j)| + |I_O(i,j) - I_O(i,j+1)|)}, (i,j) \in Z \quad (7)$$

Contrast Increase Index (CII) is defined as follows:

$$E = \frac{C_D}{C_O}, C = \frac{G_{\max} - G_{\min}}{G_{\max} + G_{\min}} \quad (8)$$

In figure 4, we compared with the Wavelet Transform based algorithm and get the evaluate number in TABLE I.

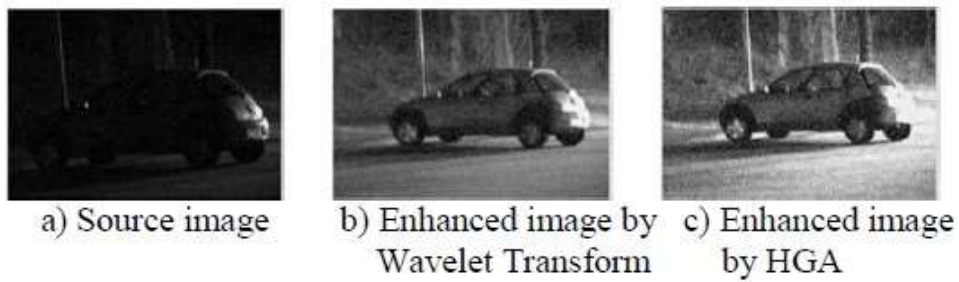


Figure 4. The Comparison of different processes

TABLE I. THE CONSTACT OF TWO METHODS

Methods	EPI	CII
Wavelet Transform	1.4567	0.7685
HGA	1.9876	3.9804

Figure 4 (a, c) show the original image and the differential evolution algorithm for enhanced results can be seen from the enhanced contrast markedly improved, clearer image details, edge feature more prominent. b, c shows the wavelet-based hybrid genetic algorithm-based Comparison of Image Enhancement: wavelet-based enhancement method to enhance image detail out some of the image visual effect is an improvement over the original image, but the enhancement is not obvious; and Hybrid genetic algorithm based on adaptive transform image enhancement effect is very good, image details, texture, clarity is enhanced compared with the results based on wavelet transform has greatly improved the image of the post-analytical processing helpful. Experimental enhancement experiment using wavelet transform "sym4" wavelet, enhanced differential evolution algorithm experiment, the parameters and the values were 5.9409, 9.5704. For a  $256 \times 256$  size image transform based on adaptive hybrid genetic algorithm in Matlab 7.0 image enhancement software, the computing time is about 2 seconds, operation is very fast. From TABLE I, objective

evaluation criteria can be seen, both the edge of the protection index, or to enhance the contrast index, based on adaptive hybrid genetic algorithm compared to traditional methods based on wavelet transform has a larger increase, which is from This section describes the objective advantages of the method. From above analysis, we can see that this method.

From above analysis, we can see that this method can be useful and effective.

## VI. CONCLUSION

In this paper, to maintain the integrity of the perspective image information, the use of Hybrid genetic algorithm for image enhancement, can be seen from the experimental results, based on the Hybrid genetic algorithm for image enhancement method has obvious effect. Compared with other evolutionary algorithms, hybrid genetic algorithm outstanding performance of the algorithm, it is simple, robust and rapid convergence is almost optimal solution can be found in each run, while the hybrid genetic algorithm is only a few parameters need to be set and the same set of parameters can be used in many different problems. Using the Hybrid genetic algorithm quick search capability for a given test image adaptive mutation, search, to finalize the transformation function from the best parameter values. And the exhaustive method compared to a significant reduction in the time to ask and solve the computing complexity. Therefore, the proposed image enhancement method has some practical value.

## REFERENCES

- [1] HE Bin et al., Visual C++ Digital Image Processing [M], Posts & Telecom Press, 2001,4:473~477
- [2] Storn R, Price K. Differential Evolution—a Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Space[R]. International Computer Science Institute, Berlaey, 1995.
- [3] Tubbs J D. A note on parametric image enhancement [J].Pattern Recognition.1997, 30(6):617-621.
- [4] TANG Ming, MA Song De, XIAO Jing. Enhancing Far Infrared Image Sequences with Model Based Adaptive Filtering [J] . CHINESE JOURNAL OF COMPUTERS, 2000, 23(8):893-896.
- [5] ZHOU Ji Liu, LV Hang, Image Enhancement Based on A New Genetic Algorithm [J]. Chinese Journal of Computers, 2001, 24(9):959-964.
- [6] LI Yun, LIU Xuecheng. On Algorithm of Image Constract Enhancement Based on Wavelet Transformation [J]. Computer Applications and Software, 2008,8.
- [7] XIE Mei-hua, WANG Zheng-ming, The Partial Differential Equation Method for Image Resolution Enhancement [J]. Journal of Remote Sensing, 2005, 9(6):673-679.

## 基于混合遗传算法的图像增强技术

**摘要**—在图像增强之中，塔布斯提出了归一化不完全  $\beta$  函数表示常用的几种使用的非线性变换函数对图像进行研究增强。但如何确定 Beta 系数功能仍然是一个问题。在图像增强处理和利用遗传算法快速算法的搜索能力进行自适应变异和搜索我们提出了一种混合遗传将微分进化算法。最后利用仿真实验证明了该方法的有效性。

**关键词**—图像增强；混合遗传算法；自适应增强

### I. 介绍

在图像形成，传递或转换过程，由于其他客观因素，如系统噪声，不足或过度曝光，相对运动等的影响会使图像通常与原始图像之间有差别（简称退化或退化）。退化图像通常模糊或信息的提取通过机器后减少甚至是错误的，它必须采取一些改进措施。

图像增强技术是在其目的是为了图像的质量这个意义上提出的。模糊图像增强情况是根据图像使用各种特殊技术集锦的一些信息图像，减少或消除不相关的信息，来强调整体或局部特征的目标图像。图像增强方法仍没有统一的理论，图像增强技术可分为三类：点运算，与空间频率增强方法增强法。本文介绍了根据图像特征自动调整自适应图像增强方法，称为混合遗传算法。为了实现图像的自适应增强它结合了差分进化自适应搜索算法，自动确定的参数值的变换函数。

### II. 图像增强技术

图像增强是图像的某些特征，如轮廓，对比，强调或突出的边缘等为了便于检测和进一步的分析和处理。增强将不会增加图像中的信息数据，但会选择适当的动态范围的功能的扩展，使得这些特点更容易检测或确定，为后续的分析处理的检测打下良好的基础。

图像增强方法包括点运算，空间滤波，频域滤波类别。点运算包括对比度拉伸，直方图建模，并限制噪声和图像减影技术。空间滤波器包括低通滤波，中值滤波，高通滤波器（锐化）。频率滤波器包括同态滤波，多尺度多分辨率图像增强中的应用[1]。

### III. 差分进化算法

差分进化（DE）首次提出了强硬的价值，并与其他进化算法进行比较，DE 算法具有强大的空间搜索能力，易实现，容易理解。DE 算法是一种新型的搜索算法，它首先是在搜索空间中随机产生初始种群，然后计算之间的任何差异向量的两个成员，所不同的添加到向量的第三个成员，通过该方法，形成一个新的个人。如果你发现新的个体成员比原来的好，然后替换原来的个体，自我的形成。

DE 操作作为遗传算法一样，它结论突变，交叉和选择，但方法是不同的。我们假设组的大小是  $P$ ，矢量维  $D$ ，我们可以表达的目标向量为（1）：

$$x_i = [x_{i1}, x_{i2}, \dots, x_{iD}] \quad (i = 1, \dots, P) \quad (1)$$

变异向量可以表示为（2）：

$$V_i = X_{r1} + F \times (X_{r2} - X_{r3}) \quad i = 1, \dots, P \quad (2)$$

$X_{r1}, X_{r2}, X_{r3}$  是三个从群中随机选择的个人，其中， $r1 \neq r2 \neq r3 \neq i$ 。F 是一系列的 [0, 2] 之间的实际类型的用于控制影响的常数因子差异向量，通常被称为比例因子。显然，矢量之间的区别越小则干扰也越小，这意味着如果组接近最佳值，扰动会自动降低。

DE 算法的选择操作是一个“贪婪”的选择模式，当且仅当新的矢量  $U_i$  比目标向量  $X_i$  更好更健全， $U_i$  将被保留到下一组。否则，目标向量  $X_i$  留在原来的组，再次作为下一代的父矢量。

#### IV. 图像增强图像的混合遗传算法

增强是获得快速对象检测的基础，因此有必要寻找实时性能好的算法。对不同系统的实际要求，许多算法需要确定的参数和人工阈值。它可以使用一个非完全 Beta 函数来完全覆盖典型变换式的图像增强，但确定 Beta 函数参数仍有许多亟待解决的问题。本节介绍了一种 Beta 功能，因为根据适用的图像增强的方法，自适应混合遗传算法的搜索的能力，自动确定变换命令的参数值来实现图像增强的自适应功能。

图像增强的目的是提高图像质量，是在指定的比较突出的特点恢复退化图像细节等。一个共同的特征的退化图像通常是对比的下侧呈明亮的，暗淡或灰色浓。低对比度退化图像可拉伸达到一种动态的直方图增强，如灰度变化。我们用  $I_{xy}$  来说明点  $(x, y)$  的灰度级它可以是由（3）表示。

$$I_{xy} = f(x, y) \quad (3)$$

其中：“f”为一个线性或非线性函数。在一般情况下，灰图像有四个非线性的翻译[6][7]，可以是如图 1 所示。我们采用归一化的 Beta 函数自动适应 4 类图像增强转变曲线。（4）中定义：

$$f(u) = B^{-1}(\alpha, \beta) \int_0^u t^{\alpha-1} (1-t)^{\beta-1} dt \quad (\alpha > 0, \beta > 0) \quad (4)$$

其中：

$$B(\alpha, \beta) = \int_0^1 t^{\alpha-1} (1-t)^{\beta-1} dt \quad (5)$$

对于不同的  $\alpha, \beta$  值，我们可以从（4）及（5）中得到响应曲线。



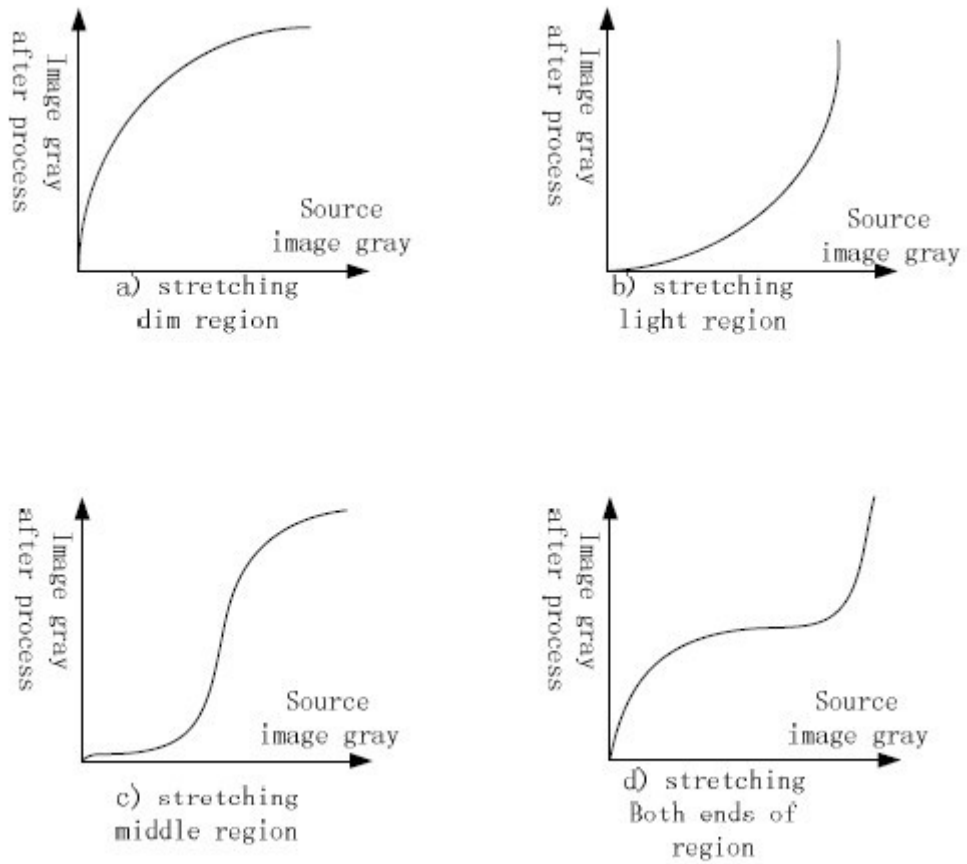


图 1 四种传统的翻译

该混合算法可以利用前面的部分自适应差分进化算法搜索最佳函数来确定的  $\beta$  值，然后每个像素灰度值为  $\beta$  函数，相应的图 1 转化，产生理想的图像增强。详细描述如下：

假设原始图像的像素  $(x, y)$  的像素的灰度水平，表示为式 (4)，记为  $i_{xy}$ ， $(x, y) \in \Omega$ ，这里  $\Omega$  是图像域。增强的图像由  $I_{xy}$  表示。首先，图像的灰度值在 (6) 中归到  $[0, 1]$ 。

$$g_{xy} = \frac{i_{xy} - i_{\min}}{i_{\max} - i_{\min}} \quad (6)$$

其中：imax 和 imin 表示图像灰度的最大值和最小值。

定义非线性变换函数  $f(U)$  ( $0 \leq U \leq 1$ ) 变换成源图像  $GXY = f(GXY)$ ，其中， $0 \leq GXY \leq 1$ 。最后，我们使用了混合遗传算法来确定适当的 Beta 函数  $f(U)$  的最佳参数  $\alpha$  和  $\beta$ 。

## V. 实验和分析

在模拟中，我们使用两种不同类型的灰度图像退化；程序执行了 50 次，人口大小为 30，进化 600 次。结果表明，提出的方法可以非常有效地提高不同退化图像类型。

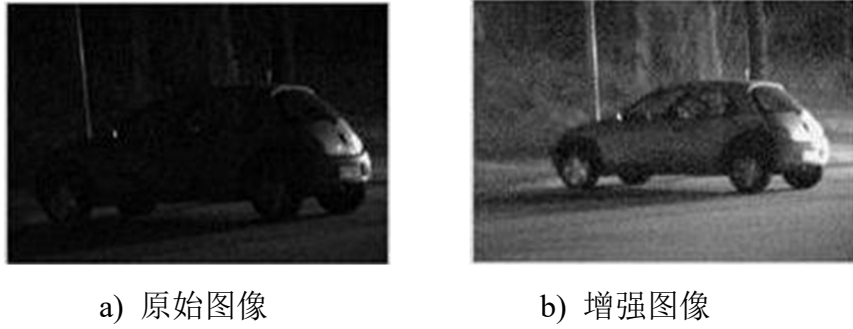


图 2 单个图像增强过程

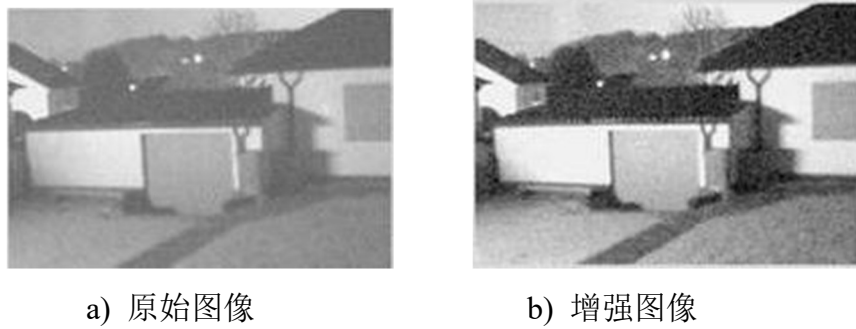


图3 移动对象增强过程

图 2，原始图像为  $320 \times 320$  的大小，它是对比度低，和更为模糊的一些细节，特别的，外围和其他细节很不明显，视觉效果差，使用文中提出的方法部分，克服了以上的一些问题，并得到令人满意的图像效果，如图 5（b）显示，该视觉效果得到明显改善。从直方图看来，图像的强度分布的范围是比较均匀，光明与黑暗的灰色区域的分布更合理了。混合遗传算法自动确定函数曲线的非线性变换，从曲线可以得出值 9.837, 5.7912，它符合图 3 的 C 级，跨越压缩变换的中间区域，这与直方图相一致，整体的原始图像低对比度，在中间区域两端压缩拉伸区域与人的视觉一致，增强效果明显提高。

图 3，原始图像的大小  $320 \times 25$ ，整体强度低，使用文中提出的方法得到 b 图像，我们可以看到地上，椅子和衣服和其他细节的分辨率和对比度比原始图像有明显改善，原始图像的灰度分布集中在较低的区域，其增强的灰度图像的灰度均匀，图 3（a）之前和之后基本的变换和非线性变换是一样的，即，图像暗区伸展的值是 5.9409, 9.5704，非线性变换的图像退化类型推断是正确的，增强视觉效果和良好的图像增强效应。

图像还没有一个统一的评价标准则很难评价图像质量的提高，有共同峰值信号噪声比（PSNR）方面的评价，但峰值信噪比不反映人类视觉系统误差。因此，我们利用边缘保护指数与对比增长指数评价实验结果。

edgel 保护指数（EPI）的定义如下（7）：

$$EPI = \frac{\sum (|I_D(i, j) - I_D(i+1, j)| + |I_D(i, j) - I_D(i, j+1)|)}{\sum (|I_O(i, j) - I_O(i+1, j)| + |I_O(i, j) - I_O(i, j+1)|)} \quad (i, j) \in Z \quad (7)$$

对比度增加指数（CII）定义如下：

$$E = \frac{C_D}{C_O}, C = \frac{G_{\max} - G_{\min}}{G_{\max} + G_{\min}} \quad (8)$$

在图 4 中，我们比较了小波变换算法得到评估表 I。



a) 原始图像

b) 通过小波变换的  
图像增强

c) 通过算法增强图像

图4 不同工艺的比较

表1 两种方法的比较

Methods	EPI	CII
Wavelet Transform	1.4567	0.7685
HGA	1.9876	3.9804

图 4 (a, c) 显示原始图像和差分进化算法增强的结果，可以看出，对比度明显提高，更清晰的图像细节，边缘特征更为突出。(B, C) 表明，基于小波变换的混合遗传算法的图像比较增强：基于小波变换的增强方法，以提高图像细节部分的视觉效果是在原始图像的改进，但增强不明显；基于自适应混合遗传算法变换的图像增强效果非常好，图像细节，纹理，清晰的结果相比提高了，基于小波变换，大大提高了图像分析后处理的能力。增强实验利用小波变换“sym4”小波，增强差分进化算法实验，它的参数和值分别为 5.9409, 9.5704。对于一个 256×256 大小的图像变换的自适应混合遗传算法在 MATLAB 7 软件中的图像增强，计算时间约为 2 秒，操作很快。从表一中客观的评价标准可以看出，无论是从边缘保护指数，或以提高对比度指数，基于自适应混合遗传算法相比传统的小波变换方法具有较大的增强，这是本节介绍的方法的客观优势。

从以上分析，我们可以看到，这种方法是有效的和有用的。

## VI. 结论

在本文中，为了保持完整性的视角的图像信息，利用混合遗传算法来进行图像增强，从实验结果可以看出，基于混合遗传算法的图像增强方法具有明显的效果。与其他进化算法相比，该算法的混合遗传算法突出表现在它是简单的，鲁棒性和快速收敛，在每次运行时发现它几乎是最佳的解决方案，该混合遗传算法只有几个参数需要设置和相同的一组参数可以用在许多不同的问题。应用混合遗传算法的快速搜索能力对于一个给定的

测试图像的自适应变异进行搜索，最终确定变换函数的最佳参数值。与穷举法相比，显着减少时间求解，解决了计算的复杂性。因此，所提出的这个图像增强方法具有一定的实用价值。

## 参考文献

- [1] HE Binetal., Visual C++ Digital Image Processing [M], Posts & Telecom Press, 2001,4:473~477
- [2] Storn R, Price K. Differential Evolution—a Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Space[R]. International Computer Science Institute, Berlaey, 1995.
- [3] Tubbs J D. A note on parametric image enhancement [J].Pattern Recognition.1997, 30(6):617-621.
- [4] TANG Ming, MA Song De, XIAO Jing. Enhancing Far Infrared Image Sequences with Model Based Adaptive Filtering [J] . CHINESE JOURNAL OF COMPUTERS, 2000, 23(8):893-896.
- [5] ZHOU Ji Liu, LV Hang, Image Enhancement Based on A New Genetic Algorithm [J]. Chinese Journal of Computers, 2001, 24(9):959-964.
- [6] LI Yun, LIU Xuecheng. On Algorithm of Image Constract Enhancement Based on Wavelet Transformation [J]. Computer Applications and Software, 2008,8.
- [7] XIE Mei-hua, WANG Zheng-ming, The Partial Differential Equation Method for Image Resolution Enhancement [J]. Journal of Remote Sensing, 2005, 9(6):673-679.

## 附录 B 主要参考文献的题录及摘要

题录[1]:

【作者】晏晓辉 朱云龙 张智聪 吕赐兴 李帅 蚁文洁

【题目】菌群优化方法及其应用研究综述

【出版社】郑州大学学报

【关键字】群体智能；优化计算；菌群优化；算法改进

【摘要】菌群优化是近年新提出的一种群体智能优化方法。在介绍几种典型菌群优化方法的基础上着重对细菌觅食算法的相关文献进行了关键词分析。结合关键词及原始细菌觅食算法的不足。从参数与结构优化“多算法混合”算子改善“多目标优化改造等 6 个方面对菌群优化方法的改进研究进行了综述。同时，对其在各典型工程领域的应用进行了介绍。最后，对菌群优化方法的发展方向进行了分析展望。

题录[2]:

【作者】郭文艳 周吉瑞 张姣姣

【题目】基于改进人工蜂群的图像增强算法

【出版社】计算机工程

【关键字】人工蜂群算法；回溯搜索算法；种群多样性；收敛速度；全局收敛；图像对比度增强

【摘要】针对人工蜂群算法易出现早熟现象和收敛速度慢等问题，提出一种基于回溯搜索的人工蜂群算法。通过回溯搜索算法选择更新种群，采用随机的变异策略和不均匀的交叉策略，增强蜂群算法种群多样性，使得改进的蜂群算法能够跳出局部最优，且具有较好的全局收敛速度。将改进的算法用于图像对比度增强，通过搜索非完全 Beta 函数的最佳参数 $\alpha, \beta$ ，确定灰度变换曲线，对图像灰度进行调整，提高图像对比度。仿真实验结果表明，该算法具有较高的求解精度和较快的收敛速度，与直方图均衡化算法相比，有效地增强了图像的对比度。

题录[3]:

【作者】刘小龙

【题目】细菌觅食优化算法的改进及应用

【出版社】华南理工大学博士学位论文

【关键字】仿生优化；细菌觅食优化算法；群体协作；生物进化；分布估计；作业调度；神经网络

【摘要】优化问题是人们在科学研究和生产实践中经常遇到的问题，人们已经对大量的最优化问题进行了深入的研究，并将其拓展成为了一门重要的学科门类。传统以梯度为基础的最速下降法、线性规划、单纯形方法等优化方法，在问题的目标函数是凸集、连续可微可导等情况下，具有较高的计算效率。但在实际应用的物流配送中心的选址、设备资源的最优分配、车间在制品的产品调度与布局等领域中，出现了许多大规模、非线性、多极值、多约束、非凸性等现象，这就使得传统的优化方法难以进行数学建模，从而给以仿生为特征的群智能计算方法提供了广阔的应用舞台，并诞生了一批模拟生物

行为的“启发式方法”，这其中的典型代表包括遗传算法 GA、蚁群优化算法 ACO、粒子群优化算法 PSO 和细菌觅食优化算法 BFO 等。相关研究表明，现有群智能算法或多或少地存在着“早熟”、“晚熟”甚至“不熟”的收敛性缺陷与问题，因此许多学者将视角瞄向了在不同的群智能算法之间取长补短，确定改进各种优化算法性能的方式与途径。

在以上主要的群体智能优化算法中，GA、ACO、PSO 都是基于高等生物作为启发对象，形成的一种“生成+检验”为特征的自适应人工智能计算技术，而 BFO 等算法，则是从微生物的行为机制出发，通过模拟细菌对环境感知的变化，而形成的一种新优化方法。由于微生物智能仿生技术问世的时间太短，国际学术界目前对 BFO 等相关研究尚有许多空白，这一新型的智能仿生算法还远未获得学术界应有的足够重视。因此，本文尝试对微生物的行为机制及其生理特性进行建模仿真，探讨这一新型智能计算方法的改进方式，从而丰富仿生优化算法中的微生物智能计算这一领域，继而对其他仿生优化算法提供一定意义上的技术借鉴，为本文提供针对现有生物体系优化算法融合改进的新途径、新视角。

本研究采用规范分析和实验研究为主的研究方法，对仿生优化的思想基础、主要门类和算法框架程式进行了探讨，阐述了仿生优化算法性能比较的问题测试函数、算法优劣的性能比较指标和算法迭代中的种群多样性度量指标，并对基本的 BFO 算法原理、实现步骤进行了深入分析，讨论了 BFO 算法中现有趋化算子、繁殖算子和迁移算子在程序执行中表现出的主要问题，进而利用测试函数对算法的各种参数设置进行了实验，提出了 BFO 算法参数设置的部分规律，基于最优觅食理论对 BFO 算法进行了实验，讨论了细菌在趋化过程中的不同觅食方式对算法性能的主要影响。

在以上理论分析的基础上，本文尝试对群智能算法进行算法思想的理论融合，试图分析现有不同群智能算法的特征，基于高等生物的群体协作、生物种群的基因进化和统计学习的分布估计三个层面，对现有算法的微观行为层面、基因改进层面和宏观指导层面进行改进，以提高 BFO 基本算法的测试性能，并使之具备协调进化和学习适应等多重智能，从而达到提高算法的搜索速度和精度的目的。本文尝试构建的改进 BFO 算法不仅具有方法上的创新，而且对现有智能计算技术具有较为积极的思想参考，从而具有一定意义上的理论创新。论文的主要研究成果表现在以下几个方面：

(1) 系统总结了仿生优化群智能计算的基本原理和主要方法，分析了现有算法性能比较的无免费午餐定理 (NFL)，阐述了算法性能比较所使用的标准测试函数、性能比较和种群多样性评价指标，揭示了各种群智能计算方法的思想基础，为后续文章所提及的算法优化及其改进研究提供评价体系和理论上的参照。

(2) 深入研究了细菌觅食优化算法的基本原理，分析了现有的趋化、繁殖和迁移算子在算法寻优中的局限性，对算法中的基本参数进行了分析，讨论了相关参数设置的经验和取值借鉴，比较了 PSO 和 BFO 中的两种觅食行为策略，分别利用细菌能量和适应度来模拟非常规和常规觅食策略，验证了不同觅食策略对算法性能的影响，最后提出了针对细菌觅食优化的四个方面的改进目标和具体改进行为策略。

(3) 分析了生物在觅食过程中的竞争和协作两种主要行为，探讨了基于协作思想的鱼群算法和学习思想的粒子群算法的主要思想。基于鱼群算法的思想，赋予细菌感知群体状态的能力，可以进行优值跟踪（追尾群体最优）和聚群（向群体中心位置好的靠拢），提出了环境感知 BFO 算法，提高了问题求解的精度。基于 PSO 的自我学习和社会学习思想，提出了协作 BFO，使得算法具有更大概率获得全局最优解。

(4) 分析了生物进化的适者生存、物种选择和遗传学说理论，讨论了基于生物进化的广义进化计算方法，分析了以进化思想为基础的遗传算法及其进化计算的基本思想。基于差分进化的思想，在细菌繁殖时通过群体内个体间的差分合作与竞争，来实现细菌群体的优化，从而对趋化周期结束后的维度退化现象进行修正，差分算子明显提高了 BFO 算法的精度、鲁棒性和全局最优获取能力。基于生命体免疫系统的思想，设置了基于免疫体的克隆繁殖算子，在趋化周期完成后对精英细菌进行克隆、高频变异和随机交叉，从而引导算子搜索，使得算法对部分测试函数具有很好的适用性，并能快速收敛，找到全局最优解。

(5) 分析了最新出现的智能计算的分布估计的方法，探讨了分布估计引入智能计算的可能性，从而可以充分利用实际问题的先验信息，完成从宏观指导思想上的建模。基于分布估计中的高斯分布思想，在细菌趋化周期结束后的繁殖环节，引入了高斯分布繁殖的概念，从宏观上对较优秀的部分细菌进行统计建模，明显提高了 BFO 算法的精度和鲁棒性，对部分测试函数具有很大的适应性。基于现有细菌繁殖的真实生长曲线，打破 BFO 算法的三层嵌套框架，模拟了细菌在优化过程中的菌群自由分布规律，建立了细菌自我繁殖和消亡的系统模型，进而从另一个侧面对前述 BFO 算法的相关性能进行佐证。

(6) 利用标准测试函数对改进细菌觅食优化的算法性能进行测试验证，在 MATLAB 软件平台上设计和开发了相应的计算机程序附后，针对实际优化中的连续空间和离散空间，采用神经网络预测问题和车间作业调度问题对算法性能进行验证，拓展了连续性 BFO 的应用空间，为神经网络的权重求解和作业调度优化提供了一种新的信息处理和智能计算工具。

#### 题录[4]:

【作者】林诗洁 董晨 陈明志 张凡 陈景辉

【题目】新型群智能优化算法综述

【出版社】计算机工程与应用

【关键字】细菌觅食优化；混合蛙跳算法；人工蜂群算法；萤火虫算法；布谷鸟搜索；果蝇优化算法；头脑风暴优化算法

【摘要】传统群智能算法在解决复杂实际多目标优化问题中存在不足，近年来学者提出诸多新型群智能算法，适用性强，在求解复杂实际问题中取得了较好的实验效果。以算法提出时间为主线，对新型群智能算法中细菌觅食优化算法、混合蛙跳算法、人工蜂群算法、萤火虫算法、布谷鸟搜索、果蝇优化算法和头脑风暴优化算法的改进及应用进行分析 and 综述，并对群智能算法未来的研究发展方向进行了探讨。

#### 题录[5]:

【作者】刘慧敏

【题目】人工鱼群算法研究及其在图像增强中的应用

【出版社】西安电子科技大学硕士学位论文

【关键字】人工鱼群算法；动态步长；行为选择；变种群策略；图像增强

【摘要】最优化方法对于解决生活中的问题有着重要的意义，也一直是众多专家学者研究的主题。比较于经典的进化算法和基于梯度特征的优化算法，群智能寻优算法表现出了良好得自组织性，简单的个体行为，优秀的全局寻优本领等诸多特点，近年来已发



展成为一个新的热门科研领域。其中，人工鱼群算法(AFSA)是起源于对鱼群追尾(AF\_Follow)、聚群(AF\_Swarm)和觅食行为(AF\_Prey)模仿的智能寻优算法，有强健的鲁棒性、优良的全局收敛性和较低的初始要求等特点，能为非线性函数的寻优研究提供较优的解决方案。但对于一些复杂情况，算法本身还是存在一些问题，例如算法寻优精度低、每次迭代耗时较长、寻优后期收敛速度慢等。

为此，文章对 AFSA 进行改善，提出采用 VPS 变种群策略的人工鱼群算法(VAFSA)。对寻优步长进行动态地调整，以提高优化的准确度；提出了具有记忆功能的行为选择策略，即人工鱼在行为选择时会首先模拟执行上次迭代的行为，若此次行为的结果优于上次行为的结果，则执行此行为，而不再模拟执行其他行为进行比较，减少了算法每一代的进化时间；采用变种群策略提高鱼群进化后期的多样性，从而提高寻优后期的收敛速度。从实验结果可以看出，VAFSA 的性能相比 AFSA 有大幅提高，算法的寻优精度更加准确，每次迭代的耗时更少，寻优后期的收敛速度更快。

研究了图像关于增强的原理，灰度变换拟合采用归一化的非完全的 Beta 函数，目标函数使用均方误差，图像增强问题被转变为寻找目标函数的最优值问题，采用 VAFSA 进行优化，提出了一种全新的图像增强算法。试验结果展示了新增强算法获得的图像具有较好的视觉效果和质量。

针对采用 VPS(Various Population Strategy)变种群策略的人工鱼群算法，如何合理地设置种群初始规模、最大规模和种群增大策略、维持策略、减小策略及此过程中人工鱼的行为选择是影响算法性能的关键。在后续科研中，将进一步钻研种群变换策略，提高 VAFSA 的性能。此外，还可以考虑 VAFSA 在其他领域的应用研究。

题录[6]:

【作者】贾占朝

【题目】遗传微粒群混合算法的研究及其在图像增强中的应用

【出版社】中南大学硕士学位论文

【关键字】微粒群算法；遗传算法；混合算法；图像增强

【摘要】微粒群算法是一种新型的、基于迭代的进化计算方法。自上世纪九十年代出现以来,由于其具有较少的参数调整以及算法简单易实现等优点,获得了国内外学者的关注和研究,它和遗传算法有相似之处,都是智能化的计算方法。遗传算法因其固有的优点,自从上世纪七十年代出现以后就受到了广泛的关注和研究。遗传算法这个经典的智能化优化算法经过多年的研究和应用,其理论已经比较完善,并且已在多个领域获得应用,取得了良好的效果。

本文分别讨论了微粒群算法和遗传算法的基本原理、流程和优缺点,在对微粒群算法和遗传算法的深入探讨后,提出了微粒群遗传混合算法。该混合算法以微粒群算法为主体,对参数进行优化。同时,为了避免由于算法初期产生微粒的随机性而消耗资源,也为了加快收敛速度,引进了摒弃因子。在对微粒根据适应度值排序之后,通过动态的摒弃因子,把适应度值较小的微粒抛弃。同时采用遗传算子来改良微粒的适应度,又结合所要解决的实际问题采用十进制编码方式,减小了时间开销。

本文对该混合算法的可行性和有效性进行了理论分析,通过对六个标准测试函数的测试,并把该混合算法的结果和标准遗传算法和微粒群算法进行比较,经由直观对比和分析,表明了该混合算法的有效性。然后,结合当前图像增强处理方面的问题,把该算法应用



于基于函数的灰度图像增强,对增强前后的图像及其对应的直方图的对比,可以看出增强效果比较明显。

题录[7]:

【作 者】王鑫 孔航 崔雨航 赵春玲 赵大众

【题 目】基于 Matlab 的图像增强算法研究

【出版社】电脑与信息技术

【关键字】图像处理; 图像增强算法; 程序代码; MATLAB 软件

【摘 要】在获取图像的过程中,由于多种因素的影响,导致图像质量会有所退化。为解决这样的问题,提高图像使用价值,图像增强处理技术应运而生。MATLAB 作为国内外流行的数字计算软件,具有强大的图像处理功能,是图像处理系统的理想开发工具。作者在 MATLAB 软件平台上,分别运用灰度均衡化、平滑滤波器、锐化滤波器、巴特沃斯低通滤波器、同态滤波器等图像处理方法进行程序处理,根据特定的需要,突出图像中的重要信息,同时减弱或去除不需要的信息,达到改善图像特征的目的。再对实验运行结果进行比较分析,发现 MATLAB 强大的运算和图形展示功能,以及简单、实用、可读性强的特点,使得这些图像增强算法变得更加的简单和直观,提高了图像处理的效率。

题录[8]:

【作 者】姜建国 周佳薇 周润生 王娟

【题 目】一种自适应细菌觅食优化算法

【出版社】控制与决策

【关键字】图像处理; 灰度图增强; 细菌觅食优化算法; 非完全 Beta 函数; 局部搜索; 位置扰动

【摘 要】为了解决现有图像增强技术在细节处理方面的不足以及变换后图像直方图分布偏移的情况,提出一种采用改进细菌觅食优化算法的灰度图像增强方法.针对细菌觅食算法在优化高维函数时性能不佳、易陷入早熟收敛的缺陷,将变高维的灰度图像增强问题转化为固定 2 维的非完全 Beta 函数的参数最优化问题.仿真实验结果表明了所提出方法的有效性,与其他方法相比,增强后的图像细节表现更自然,直方图分布更均匀,明暗区域分配更合理。

题录[9]:

【作 者】Dong Hwa Kim Ajith Abraham Jae Hoon Cho

【题 目】A hybrid genetic algorithm and bacterial foraging approach for global optimization

【出版社】Information Sciences

【关键字】Genetic algorithm; Bacterial foraging optimization; Hybrid optimization; Controller tuning

【摘 要】The social foraging behavior of Escherichia coli bacteria has been used to solve optimization problems. This paper proposes a hybrid approach involving genetic algorithms (GA) and bacterial foraging (BF) algorithms for function optimization problems. We first illustrate the proposed method using four test functions and the performance of the algorithm is studied with an emphasis on mutation, crossover, variation of step sizes, chemotactic steps, and the lifetime of the bacteria. The proposed algorithm is then used to tune a PID controller

of an automatic voltage regulator (AVR). Simulation results clearly illustrate that the proposed approach is very efficient and could easily be extended for other global optimization problems.

题录[10]:

【作 者】 Jun Sun Wei Fang Vasile Palade Xiaojun Wu Wenbo Xu

【题 目】 Quantum-behaved particle swarm optimization with Gaussian distributed local attractor point

【出版社】 Applied Mathematics and Computation

【关键字】 Particle swarm optimization;Gaussian probability distribution;Swarm intelligence;Quantum behavior

【摘 要】 This paper proposes a novel variant of quantum-behaved particle swarm optimization (QPSO) algorithm with the local attractor point subject to a Gaussian probability distribution (GAQPSO). The local attractor point in QPSO plays an important in that determining the convergence behavior of an individual particle. As such, the mean value and standard deviation of the proposed Gaussian probability distribution in GAQPSO are carefully selected. The distributions and diversities of the local attractor points in GAQPSO and QPSO are evaluated and compared. For the purpose of comparison, two variants of the GAQPSO algorithm are proposed by using a mutation probability and other types of probability distribution. The GAQPSO has been comprehensively evaluated on the suite of CEC2005 benchmark functions, and the experimental results are compared with those of the PSO and QPSO algorithms based on different probability distributions. It is shown by the results that the GAQPSO algorithm is an effective approach that can improve the QPSO performance considerably, that is, the GAQPSO algorithm is less likely to be stuck in local optima and hence it can achieve better solutions in most cases.

## 附录 C 主要源码

程序主要源码：

mybfo.m

---

```
%*****细菌觅食优化算法图像增强*****
clc; clear; close all
warning off
feature jit off    % 加速代码执行

Img=imread('lena.jpg');
%Img=rgb2gray(Img);

[M,N]=size(Img);
I1=double(Img);
lmax=double(max(max(Img)));
lmin=double(min(min(Img)));
I2=(I1-lmin)/(lmax-lmin);          % 图像归一化处理

%%%%%%%%%%%%BFO 算法-----%%%%%%%%%%%%
%-----BFO 初始化参数-----
bounds = [0,10;0,10]; % 函数变量范围
d = 2;          % 搜索范围的维度
s = 26;         % 细菌的个数
Nc = 5;         % 趋化的次数
Ns = 5;         % 趋化操作中单向运动的最大步数
C(:,1) = 0.2*ones(s,1); % 翻转选定方向后，单个细菌前进的步长
Nre = 5;        % 复制操作步骤数
Ned = 5;        % 驱散(迁徙)操作数
Sr = s/2;       % 每代复制（分裂）数
Ped = 0.25;     % 细菌驱散(迁徙)概率
d_attract = 0.05; % 吸引剂的数量
ommiga_attract = 0.05; % 吸引剂的释放速度
h_repellant = 0.05; % 排斥剂的数量
ommiga_repellant = 0.05; % 排斥剂的释放速度
% 产生初始细菌个体的位置
P(1,:,1,1) = 10*rand(1,s);
P(2,:,1,1) = 10*rand(1,s);
```

```
%引入 PSO 参数 改进
```

```
c1start=1.49445;
```

```
c2start=1.49445;
```

```
Pbesttable=zeros(2,s);
```

```
JPbesttable=inf(1,s);
```

```
gbest=zeros(2,1);
```

```
%----细菌趋药性算法循环开始
```

```
%---- 驱散(迁徙)操作开始
```

```
for ll = 1:Ned
```

```
    ll
```

```
    %-----复制操作开始
```

```
    for kk = 1:Nre
```

```
        kk
```

```
        %-----趋化操作(翻转或游动)开始
```

```
        for jj = 1:Nc
```

```
            %-----对每一个细菌分别进行以下操作
```

```
            for ii = 1:s
```

```
                %-----计算函数 J(ii,jj,kk,ll)，表示第 i 个细菌在第 l 次驱散第 k 次
```

```
                %-----复制第 j 次趋化时的适应度值
```

```
                J(ii,jj,kk,ll) = fitness_bfo(P(:,ii,jj,kk,ll),I2);
```

```
                %-----修改函数，加上其它细菌对其的影响
```

```
                Jcc = sum(-d_attract*exp(-ommiga_attract*((P(1,ii,jj,kk,ll)-...  
                    P(1,1:26,jj,kk,ll)).^2+(P(2,ii,jj,kk,ll)-P(2,1:26,jj,kk,ll)).^2))+...  
                    sum(h_repellant*exp(-ommiga_repellant*((P(1,ii,jj,kk,ll)-...  
                    P(1,1:26,jj,kk,ll)).^2+(P(2,ii,jj,kk,ll)-P(2,1:26,jj,kk,ll)).^2)));
```

```
                J(ii,jj,kk,ll) = J(ii,jj,kk,ll) + Jcc;
```

```
                %----保存细菌目前的适应度值，直到找到更好的适应度值取代之
```

```
                Jlast = J(ii,jj,kk,ll);
```

```
            if J(ii,jj,kk,ll)<JPbesttable(ii)
```

```
                %记录该细菌的个体历史最优位置 以及对应的适应度
```

```
                JPbesttable(ii)=J(ii,jj,kk,ll);
```

```
                Pbesttable(:,ii)=P(:,ii,jj,kk,ll);
```

```
            end
```

```
        %找到全局最优适应度
```

```
        [JPaa,JPbb]=find(JPbesttable==min(JPbesttable));
```

```
        %找到全局最优的细菌位置 (JPbesttable 可能有相同的最优值，JPbb  
        的值可能有 1 个以上，取第一个)
```

```
        gbest=Pbesttable(:,JPbb(1));
```

```

%          %-----翻转，产生一个随机向量 C(i),代表翻转后细菌的方向
%          Delta(:,ii) = (2*round(rand(d,1))-1).*rand(d,1);
%          % PHI 表示翻转后选择的一个随机方向上前进
%          PHI = Delta(:,ii)/sqrt(Delta(:,ii)'*Delta(:,ii));
%          PHIpbest=(P(:,ii,jj,kk,ll)-Pbesttable(:,ii))/...

sqrt((P(:,ii,jj,kk,ll)-Pbesttable(:,ii))*(P(:,ii,jj,kk,ll)-Pbesttable(:,ii)));
PHIgbest=(P(:,ii,jj,kk,ll)-gbest)/...
sqrt((P(:,ii,jj,kk,ll)-gbest)*(P(:,ii,jj,kk,ll)-gbest));
if sum(isnan(PHIpbest))>=1
    PHIpbest=[0;0];
end
if sum(isnan(PHIgbest))>=1
    PHIgbest=[0;0];
end
w=sin((pi/2)*(Nc-jj)/Nc);
c1=c1start*cos(pi+jj/Nc*pi);
c2=c2start*cos(pi+jj/Nc*pi);
r1=rand();
r2=rand();
PHIrand=-pi/4+pi/2*rand(2,1);
PHI=w*PHIrand+c1*r1*PHIpbest+c2*r2*PHIgbest;
%-----移动，向着翻转后细菌的方向移动一个步长，并且改变细菌的
位置
P(:,ii,jj+1,kk,ll) = P(:,ii,jj,kk,ll) + C(ii,kk)*PHI;
%边界限制 分别用 最小值+rand 和 最大值-rand 代替
[aa,bb]=find(P(:,ii,jj+1,kk,ll)<bounds(:,1));
P(aa,ii,jj+1,kk,ll)=bounds(aa,1)+rand(length(aa),1);
[aa,bb]=find(P(:,ii,jj+1,kk,ll)>bounds(:,2));
P(aa,ii,jj+1,kk,ll)=bounds(aa,2)-rand(length(aa),1);

%-----计算细菌当前位置的适应度值
J(ii,jj+1,kk,ll) = fitness_bfo(P(:,ii,jj+1,kk,ll),I2);
%-----游动-----
m = 0;          % 给游动长度计数器赋初始值
while(m < Ns) % 未达到游动的最大长度，则循环
    m = m + 1;
    % 新位置的适应度值是否更好？如果更好，将新位置的适应度
值
    % 存储为细菌 i 目前最好的适应度值
    if(J(ii,jj+1,kk,ll)<Jlast)

```

```

Jlast = J(ii,jj+1,kk,ll); % 保存更好的适应度值

%记录该细菌的个体历史最优位置
Pbesttable(:,ii)=P(:,ii,jj+1,kk,ll);
JPbesttable(ii)=Jlast;

% 在该随机方向上继续游动步长单位,修改细菌位置
P(:,ii,jj+1,kk,ll) = P(:,ii,jj+1,kk,ll) + C(ii,kk)*PHI;

%边界限制 分别用 最小值+rand 和 最大值-rand 代替
[aa,bb]=find(P(:,ii,jj+1,kk,ll)<bounds(:,1));
P(aa,ii,jj+1,kk,ll)=bounds(aa,1)+rand(length(aa),1);
[aa,bb]=find(P(:,ii,jj+1,kk,ll)>bounds(:,2));
P(aa,ii,jj+1,kk,ll)=bounds(aa,2)-rand(length(aa),1);

% 重新计算新位置上的适应度值
J(ii,jj+1,kk,ll) = fitness_bfo(P(:,ii,jj+1,kk,ll),I2);
else
    % 否则，结束此次游动
    m = Ns;
end
end
J(ii,jj,kk,ll) = Jlast; % 更新趋化操作后的适应度值

end % 如果 i<N, 进入下一个细菌的趋化, i=i+1
%-----如果 j<Nc, 此时细菌还处于活跃状态, 进行下一次趋化,
j=j+1----->Jlast
x = P(1, :, jj, kk, ll);
y = P(2, :, jj, kk, ll);
clf
plot(x,y,'*','markers',6) % 绘图
% set(gcf,'color',[1,1,1])
axis([0 10 0 10]); % 设置图的坐标
pause(.01) % 暂停 0.1 秒后继续
hold on;
end
%---下面进行复制操作
%-----复制-----
%-----根据所给的 k 和 l 的值, 将每个细菌的适应度值按升序排序
Jhealth = sum(J(:, :, kk, ll), 2); % 给每个细菌设置健康函数值
[Jhealth, sortind] = sort(Jhealth); % 按健康函数值升序排列函数
P(:, :, 1, kk+1, ll) = P(:, sortind, Nc+1, kk, ll);

```

```

C(:,kk+1) = C(sortind, kk);
%-----将代价小的一半细菌分裂成两个，代价大的一半细菌死亡
for ii = 1: Sr
    % 健康值较差的 Sr 个细菌死去，Sr 个细菌分裂成两个子细菌，保持个体
    总数的 s 一致性
    P(:,ii+Sr,1, kk+1, ll) = P(:,ii,1, kk+1, ll);
    C(ii+Sr, kk+1) = C(ii, kk+1);
end
%-----如果 k<Nre，转到(3)，进行下一代细菌的趋化
end
%-----趋散，对于每个细菌都以 Ped 的概率进行驱散，但是驱散的细菌群体的总数
%-----保持不变，一个细菌被驱散后，将被随机重新放置到一个新的位置
for m = 1:s
    % 产生随机数，如果既定概率大于该随机数，细菌 i 灭亡，随机产生新的细菌
i
    if(Ped > rand)
        P(:,m,1,1,1) = 10*rand(1,d);
    else
        P(:,m,1,1,ll+1) = P(:,m,1,Nre+1,ll); % 未驱散的细菌
    end
end
end % 如果 l<Ned，转到(2)，否则结束
% 输出最优参数值
reproduction = J(:,1:Nc,Nre,Ned); % 每个细菌最小的适应度值
[Jlastreproduction,O] = min(reproduction,[],2);
[BestY,I] = min(Jlastreproduction);
Pbest = P(:,I,O(I,:),kk,ll);
disp('Best solution:')
display(['x = ' mat2str(Pbest(1),2)])
display(['y = ' mat2str(Pbest(2),2)])

% 得到最优非完全 beta 函数参数后进行变换再反归一化得到增强图像
I3=betainc(I2,Pbest(1),Pbest(2));% 用非完全 beta 函数对归一化后图像作自适应变换
I4=round((lmax-lmin).*I3+lmin); % 反归一化处理
I5=uint8(I4); % 得到增强后图像
% 输出图像
figure;
subplot(2,3,1),imshow(Img),title('原图');
I6=histeq(Img);
subplot(2,3,2),imshow(I6),title('直方图均衡化');
subplot(2,3,3),imshow(I5),title('BFO 改进增强图像');
subplot(2,3,4),imhist(Img),title('原图直方图');

```

```
subplot(2,3,5),imhist(I6),title('直方图均衡化后直方图');
subplot(2,3,6),imhist(I5),title('BFO 改进增强后直方图');
```

#### fitness\_bfo.m

```
% 图像适应度函数
function fit=fitness_bfo(sol,Img)
a=sol(1);
b=sol(2);
[M,N]=size(Img);
n=M*N;
I1=uint8(Img);
lmax=double(max(max(I1)));
lmin=double(min(min(I1)));
I3=betainc(Img,a,b);    % 用非完全 beta 函数对归一化后图像作自适应变换
%I4=round(255*((lmax-lmin).*I3+lmin));    % 反归一化处理
I4=round(255.*I3);
I5=uint8(I4);    % 得到增强后图像
Fac=sum(sum(I5.^2))/n-(sum(sum(I5))/n).^2;    % 求方差
E=entropy(I5);    % 图像信息熵

% Fbr 像素差别
Fbr=0;
for ii=1:M-2
    Ix=I5(ii,:)-I5(ii+2,:);
    Fbr=Fbr+sum(Ix.*Ix);
end

%P  周长
P=0;
%公式右半部分
for ii=1:M-1
    Px=I5(ii,:)-I5(ii+1,:);
    P=P+sum(abs(Px));
end
%公式左半部分
for jj=1:N-1
    Py=I5(:,jj)-I5(:,jj+1);
    P=P+sum(abs(Py));
end
```



% A 面积

A=sum(sum(I5));

%紧致度 C

C=P\*P/A;

% 信噪改变量

Inc=n;

% t=127;

% [aa,bb]=find(I5>t);

% Inc=length(aa);

% 设计的适应度函数

fit=-E\*Inc\*(Fac+2.5\*C)+Fbr;