

多融合策略改进灰狼优化算法的图像分割

古力加依娜·木合亚提, 姑丽加玛丽·麦麦提艾力*

新疆师范大学数学科学学院, 新疆 乌鲁木齐

收稿日期: 2024年3月29日; 录用日期: 2024年4月19日; 发布日期: 2024年4月30日

摘要

本文提出了一种多策略融合改进的灰狼优化算法的图像分割。首先, 针对灰狼优化算法的收敛速度慢, 易陷入局部最优解等问题分别采用Tent混沌映射优化、收敛因子非线性调整、透镜成像反向学习、修改位置更新方程策略来提升灰狼算法的搜索性能和收敛速度, 并通过不同的基准测试函数验证了改进算法的优越性。其次, 通过改进的灰狼优化算法获取K均值聚类算法的初始聚类中心, 并应用于图像分割当中从而进一步提高图像分割效率。最后, 将所提出的方法应用于医学图像进行实验。实验结果表明, 与传统的图像分割算法和其他优化算法相比, 本文提出的方法明显提高了图像分割的质量和效果, 且具有更好的鲁棒性和稳定性。

关键词

图像分割, 灰狼优化算法, 镜像反向学习, Tent混沌映射优化, K均值聚类算法

Gray Wolf Optimization Algorithm for Image Segmentation Improved by Multiple Fusion Strategies

Gulijayina Muheyati, Gulijamali Maimaitiaili*

College of Mathematical Sciences, Xinjiang Normal University, Urumqi Xinjiang

Received: Mar. 29th, 2024; accepted: Apr. 19th, 2024; published: Apr. 30th, 2024

Abstract

This paper presents a multi-strategy integrated enhancement of the gray wolf optimization algorithm for image segmentation. Initially, various strategies such as Tent chaotic mapping optimization, non-linear adjustment of convergence factor, reverse-learning based lens imaging, and posi-

*通讯作者。

文章引用: 古力加依娜·木合亚提, 姑丽加玛丽·麦麦提艾力. 多融合策略改进灰狼优化算法的图像分割[J]. 图像与信号处理, 2024, 13(2): 225-237. DOI: 10.12677/jisp.2024.132019

tion updating are employed to enhance the search performance and convergence speed of the gray wolf algorithm, addressing issues like slow convergence speed and susceptibility to local optima. The enhanced algorithm is validated through different benchmark test functions. Subsequently, the improved gray wolf optimization algorithm is utilized to obtain initial clustering centers for image segmentation, further enhancing the efficiency of the segmentation process. Finally, the proposed method is applied to medical images in experimental settings. The experimental results demonstrate that compared to traditional image segmentation algorithms and other optimization algorithms, the proposed method significantly improves the quality and effectiveness of image segmentation, exhibiting better robustness and stability.

Keywords

Image Segmentation, Gray Wolf Optimization Algorithm, Reverse-Learning Based Lens Imaging, Tent Chaotic Mapping, Kmeans Clustering Algorithm

Copyright © 2024 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

图像分割作为图像处理研究中的热点之一, 对于计算机视觉和模式识别起着重要的作用。图像分割已广泛应用于医学图像处理[1]、自动驾驶[2]、目标检测和跟踪、地质勘探[3]和环境监测等多个领域。图像分割的目的是根据图像中的不同特征或属性, 将其划分为具有相似纹理、颜色、亮度、对比度、形状和大小的均匀部分或类, 从而为后续的图像处理和分析提供基础。图像分割作为图像处理领域的基础任务, 其质量直接影响到后续的图像分析和应用。因此, 图像分割一直是图像处理的研究热点之一。

K 均值聚类算法[4]图像分割是一种常用的图像分割方法。它基于聚类的思想, 将一幅图像中的像素分为不同的类别或者聚类簇, 使得同一类别内的像素具有相似的特征, 而不同类别之间的特征差异最大化。Kmeans 聚类算法图像分割的优点在于简单、高效, 且易于理解和实现。然而, 该算法需要事先随机初始化聚类中心, 可能会受到初始聚类中心选择的影响, 导致结果不稳定且易于陷入局部最优解。因此, 为了克服这些问题许多学者将 Kmeans 聚类算法跟智能优化算法相结合, 提出了改进的图像分割算法。在文献[5]中用改进的正余弦算法来优化传统 Kmeans 聚类算法, 以改善图像分割效果。在文献[6]中动态粒子群优化和 Kmeans 聚类算法相结合提出了 DPSOK 算法, 该算法既保留了聚类算法收敛速度快的优点, 又克服了其易陷入局部最优的缺点。在文献[7]中利用改进的蝠鲞觅食优化算法优化 Kmeans 聚类算法, 优化后的图像分割方法提高了分割质量及泛化能力。

灰狼优化算法[8] (Grey Wolf Optimizer, GWO)是一种元启发式优化算法, 基于自然界中灰狼的领导等级和群体狩猎机制提出。GWO 算法原理简单、参数少、编程方便、支持分布式并行计算、全局搜索能力强等优点, 在工程问题、机器学习、医学和生物信息学等许多领域中得到广泛应用。然而, 与其他智能算法类似, 灰狼优化算法也存在收敛速度慢、收敛精度低、容易陷入局部最优的缺点。出于这个原因, 许多研究人员改进了灰狼优化算法。在文献[9]中将生物进化和自然生物更新的“适者生存”原则引入到灰狼算法中, 提出了一种具有演化和消除机制的改进灰狼优化器, 进一步加速了灰狼优化算法的收敛速度, 提高了优化精度。在文献[10]为提高灰狼的搜索能力, 基于随机游走的思想, 通过引入随机性来增加灰狼搜索空间的多样性, 进而改善算法的性能。在文献[11]中为解决基于群体的元启发式优化算法中参数调整和停止标准缺乏系统性的问题, 基于候选解决方案的适应度历史来自适应调整探索/开发

参数, 提出了自适应灰狼优化算法, 来提高了算法的性能和效率。在文献[12]中灰狼优化算法中引入了迁移操作, 并根据灰狼的适应度函数值动态修改迁移概率, 并将其应用于露天矿的低碳运输调度。

受以上文献的启发, 基于 GWO 的特点和优势, 本文提出了一种基于改进灰狼优化算法的 Kmeans 图像分割方法, 旨在提高 K-means 算法在图像分割中的性能。

2. 多融合策略改进的灰狼算法

2.1. 灰狼优化算法

Mirjalili [8]等学者分析灰狼的社会行为, 对灰狼的社会行为和狩猎过程进行数学建模, 设计了灰狼优化算法。灰狼算法的数学模型由以下几部分组成:

灰狼的包围猎物的数学模型如下[8]:

$$X(t+1) = X_p(t) - A \cdot D \quad (1)$$

其中 t 为当前迭代次数; $X_p(t)$ 与 $X(t)$ 分别为猎物与灰狼当前位置, $X(t+1)$ 为下一个位置; A 为系数矩阵; D 为系数向量。它们是控制勘探与开发衔接与转换的重要参数。 D 由食饵的位置决定, 可以根据下面的公式计算:

$$D = |C \cdot X_p(t) - X(t)| \quad (2)$$

$$C = 2 \cdot r_2 \quad (3)$$

式中: r_2 为区间 $[0,1]$ 随机生成的向量。

$$A = 2a \cdot r_1 - a \quad (4)$$

$$a = 2 - \frac{2t}{t_{\max}} \quad (5)$$

式中: t_{\max} 为最大迭代次数; t 为当前迭代次数; a 随着迭代次数的增加从 2 线性递减到 0; r_1 是从区间 $[0,1]$ 随机生成的向量, 使得矩阵 A 中的数值被限制在一个区间 $[-2a, 2a]$ 内。其他狼只需要更新自己的位置, 计算公式如下[8]:

$$X(t+1) = \frac{X_1 + X_2 + X_3}{3} \quad (6)$$

其中 X_1, X_2 和 X_3 的计算公式如下:

$$\begin{aligned} X_1 &= X_\alpha - A_1 \cdot D_\alpha \\ \bar{X}_2 &= X_\beta - A_2 \cdot D_\beta \\ \bar{X}_3 &= X_\delta - A_3 \cdot D_\delta \end{aligned} \quad (7)$$

$$\begin{aligned} D_\alpha &= |C_1 \cdot X_\alpha - X| \\ D_\beta &= |C_2 \cdot X_\beta - X| \\ D_\delta &= |C_3 \cdot X_\delta - X| \end{aligned} \quad (8)$$

2.2. Tent 混沌映射优化初始种群策略

在 GWO 中, 初始种群的选择对算法的性能和收敛速度有一定的影响。标准的 GWO 种群初始化方法通常使用随机生成的数值, 但这种方法可能会导致初始种群的分布较为均匀, 以及种群中个体之间的

相似度较高, 从而降低了搜索的多样性和探索能力。然而, 混沌映射是一种具有混沌行为的非线性动力系统, 具有高度敏感性和随机性, 可以生成一系列具有较高随机性和多样性的数值。利用混沌映射初始化 GWO 的初始种群能提高 GWO 的全局搜索能力。因此, 本文采用 Tent 混沌映射来生成初始种群, 目的是提高灰狼的多样性。Tent 混沌映射[13]的数学定义如下:

$$z_{k+1} = \begin{cases} z_k/\beta & z_k \in (0, \beta] \\ (1-z_k)/(1-\beta) & z_k \in (\beta, 1] \end{cases} \quad (9)$$

式中, $\beta \in (0, 1)$ 。本文取 $\beta = 0.5$ 。

对采用的 Tent 混沌映射初始化的 GWO_1 算法与随机初始化的 GWO 算法进行实验对比以验证改部分改进的有效性, 如表 1 所示:

Table 1. Experimental comparison of population initialization.

表 1. 初始化种群实验对比

f	GWO	GWO_1	f	GWO	GWO_1
f_1	1.03E-29	3.58E-30	f_4	5.66E-14	8.55E-14
f_2	1.30E-17	1.89E-17	f_5	7.90E-14	7.54E-14
f_3	4.63E-08	6.04E-09	f_6	5.46E-3	4.62E-3

从表 1 中可知, 在相同的测试环境下除了 f_5 和 f_6 的值略差于 GWO, 其他测试函数上 GWO_1 的求解精度高于 GWO 的求解精度。

2.3. 收敛因子非线性调整策略

GWO 算法通过参数 a 和 A 的自适应值来探索和利用。较大的 a 值有助于进行全局搜索, 而较小的 a 值则有助于进行局部搜索。适当选择 a 参数可以在全局和局部探索能力之间提供平衡。然而, 根据等式 (5), 收敛因子 a 随着迭代次数从 2 到 0 线性下降, 但灰狼个体在搜索猎物的过程中并不是线性变化的, 是非线性的、高度复杂的过程。因此, 收敛因子 a 的线性下降并不能完全反映实际的优化搜索过程。因此, 为了平衡 GWO 算法的探索能力和开发能力, 许多学者提出了控制参数 a 随迭代次数非线性变化[14][15]。本文中应用的非线性控制参数[15], 如下式所示:

$$a' = 2 \left(1 - \left(\frac{t}{T_{\max}} \right)^2 \right) \quad (10)$$

式中: t 为当前迭代次数, T_{\max} 为最大迭代次数。

从图 1 中可以看出, 改进后的 a' 更能反映实际的优化搜索过程。

对采用改进非线性收敛因子的 GWO_2 算法与标准的 GWO 算法进行实验对比以验证改部分改进的有效性, 如表 2 所示:

Table 2. Experimental comparison of convergence factors.

表 2. 收敛因子的实验对比

f	GWO	GWO_2	f	GWO	GWO_2
f_1	1.03E-29	8.32E-37	f_4	5.66E-14	7.57E-15
f_2	1.30E-17	1.17E-21	f_5	7.90E-14	2.22E-14
f_3	4.63E-08	7.97E-10	f_6	5.46E-3	8.20E-4

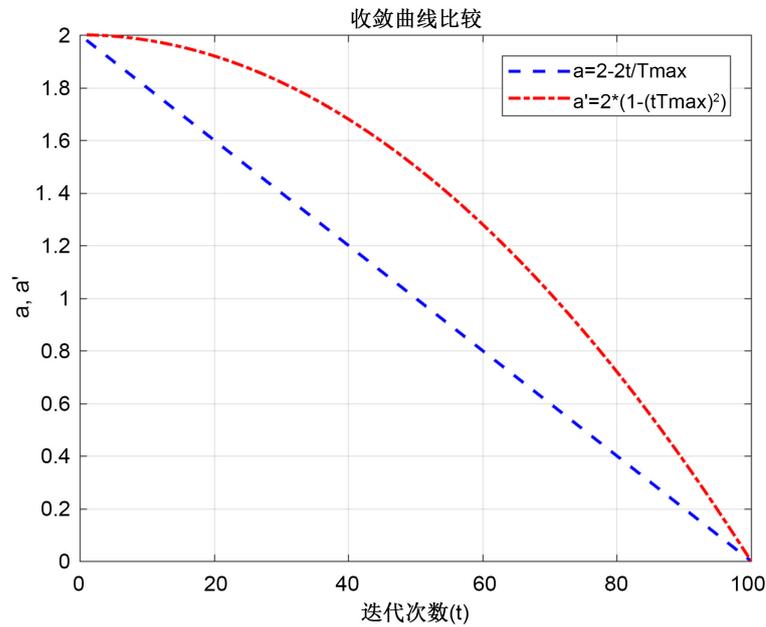


Figure 1. The convergence curve of parameter a
图 1. 参数 a 的收敛曲线

从表 2 中可知, 在相同的测试环境下除了 f_5 的值逊于 GWO, GWO_2 的求解精度明显高于 GWO 的求解精度。说明采用非线性收敛因子, 与线性收敛因子相比会有很好的效果, 这也说明了非线性收敛因子的有效性。

2.4. 透镜成像反向学习策略

透镜反向学习是一种新颖的策略, 它将反向学习机制与透镜成像原理相结合, 其基本思想是同时计算和比较候选解和对应的反向解, 通过与相反解的比较, 选择最佳候选解开始迭代计算。在文献[16]中在 GWO 中引入透镜反向学习策略能够扩大了算法群体搜索的有效范围, 从而降低了算法陷入局部最优的概率, 助算法跳出局部最优。具体过程如下[16]:

$$x_j^{**} = \frac{a_j + b_j}{2} + \frac{a_j + b_j}{2k} - \frac{x_j^*}{k} \quad (11)$$

式中: x_j^* 和 x_j^{**} 分别为 x^* 和 x^{**} 的第 j 维分量, a_j 和 b_j 分别为决策变量的上下界的第 j 维分量。

对采用透镜反向学习策略的 GWO_3 算法与标准的 GWO 算法进行实验对比以验证改部分改进的有效性, 如表 3 所示:

Table 3. Experimental comparison of lens inverse learning strategies.
表 3. 透镜反向学习实验对比

f	GWO	GWO_3	f	GWO	GWO_3
f_1	1.03E-29	0	f_4	5.66E-14	0
f_2	1.30E-17	8.95E-216	f_5	7.90E-14	8.88E-16
f_3	4.63E-08	0	f_6	5.46E-3	0

从表 3 中可知, 在相同的测试环境下, f_2 和 f_5 上没有得到理论最优解外, 但是求解精度高于 GWO,

在其他函数上 GWO_3 达到理论最优解。说明与传统 GWO 算法相比, GWO_3 算法能够更好地克服局部最优解问题, 提高搜索精度和收敛速度。

2.5. 位置更新策略

由式(6)的位置更新方程可以判断, WOA 考虑了个体位置和最优种群位置, 以及个体与种群之间的信息交换, 但忽略了个体经验对信息交换的影响。因此, 在位置更新调整策略中[17]引入 PSO 中的个体速度和位置更新, 以便个体保存其最优位置信息。灰狼追踪猎物位置的数学模型可以修改为[17]:

$$X_i(t+1) = c_1 r_1 (w_1 X_1(t) + w_2 X_2(t) + w_3 X_3(t)) + c_2 r_2 (X_{ibest} - X_i(t)) \quad (12)$$

$$\begin{aligned} w_1 &= \frac{X_1}{X_1 + X_2 + X_3} \\ w_2 &= \frac{X_2}{X_1 + X_2 + X_3} \\ w_3 &= \frac{X_3}{X_1 + X_2 + X_3} \end{aligned} \quad (13)$$

式中: c_1, c_2 分别为社会学习因子和认知学习因子(本文中取 $c_1 = c_2 = 2$)。它们分别代表个体最优值和群体最优值的影响; \vec{X}_{ibest} 表示灰狼个体本身经历过最优的位置; w_1, w_2, w_3 为惯性权重系数。

对采用新的位置更新 GWO_4 算法与标准的 GWO 算法进行实验对比以验证改部分改进的有效性, 如表 4 所示:

Table 4. Position update strategy comparative Experiment

表 4. 位置更新策略的对比实验

f	GWO	GWO_4	f	GWO	GWO_4
f_1	1.03E-29	4.93E-317	f_4	5.66E-14	0
f_2	1.30E-17	2.01E-201	f_5	7.90E-14	4.44E-15
f_3	4.63E-08	5.95E-221	f_6	5.46E-3	0

从表 4 中可知, 在相同的测试环境下, GWO_4 算法能精确求出 f_5, f_6 的理论最优解, 在其他测试函数上的求解精度均高于 GWO。说明与标准的 GWO 相比 GWO_4 的求解精度明显提升。

2.6. 改进灰狼优化算法的测试

综合上述改进思想, 改进灰狼优化算法(IGWO)算法的实现步骤如下:

Step1 输入灰狼种群 N ; 最大迭代次数 t_{max} ; 空间维度 D ; 搜索范围 $[lb, ub]$;

Step2 利用 Tent 混沌映射(9)初始化灰狼种群;

Step3 根据寻优函数计算每个灰狼个体的适应度 $f(X_\alpha)$, 并按照适应度值进行排序, 选择前三个最好的狼, 记录其位置 X_α, X_β 和 X_δ ;

Step4 利用式(11)对 X_α 进行透镜成像反向学习策略, 产生反向解 X'_α 并计算 X'_α 的适应度 $f(X'_\alpha)$;

Step5 判断 $f(X_\alpha) < f(X'_\alpha)$ 若满足, 将反向解 X'_α 替换为 X_α 加入种群参与迭代;

Step6 根据式(10)计算衰减因子 a' ;

Step7 根据式(3) (4)计算参数 A, C ;

Step8 根据式(12)更新每个灰狼的位置;

Step9 判断是否满足终止条件, 若满足, 则运算结束, 输出最优解; 否则, 令 $t = t + 1$, 返回 Step5 挑

至步骤 5 直到计算达到最大迭代次数。

为了验证 IGWO 算法的性能, 本文选取了 6 个基准测试函数来验证所提出算法的寻优性能。测试函数如表 5 所示, 其中 $f_1 \sim f_3$ 是单峰值函数, $f_4 \sim f_6$ 是多峰值函数。提出的 IGWO 算法与其他四种算法进行比较, 即原始的 GWO 算法[8], 鲸鱼优化算法[18] (Whale optimization algorithm, WOA), 麻雀优化算法[19] (Sparrow Search Algorithm, SSA)和粒子群优化算法[20] (Particle Swarm Optimization, PSO)。为了进行公平比较, 5 种算法在相同的种群规模和迭代次数下进行实验, 即种群规模为 30, 迭代次数为 500。

Table 5. Benchmark test functions

表 5. 基准测试函数

函数表达式	维数	区间	最优解
$f_1(x) = \sum_{i=1}^n x_i^2$	30	[-100, 100]	0
$f_2(x) = \sum_{i=1}^n x_i + \prod_{i=1}^n x_i $	30	[-10, 10]	0
$f_3(x) = \sum_{i=1}^n \left(\sum_{j=1}^i x_j \right)^2$	30	[-100, 100]	0
$f_4(x) = \sum_{i=1}^n \left[x_i^2 - 10 \cos(2\pi x_i) + 10 \right]$	30	[-5.12, 5.12]	0
$f_5(x) = -20 \exp \left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2} \right) - \exp \left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i) \right) + 20 + e$	30	[-100, 100]	0
$f_6(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos \left(\frac{x_i}{\sqrt{i}} \right) + 1$	30	[-600, 600]	0

所有测试均独立运行 30 次。实验中结果包括最佳值(Best)、最差值(Worst)、平均值(Mean)和标准差(Std)。实验结果如表 6, 表 7 所示。

Table 6. Comparison of test results on Single-peak function

表 6. 单峰函数上的测试结果对比

Fun	Algorithm	Best	Worst	Mean	Std
f_1	GWO	2.6867e-29	4.3984e-27	8.9291e-28	1.1261e-27
	WOA	1.1478e-31	2.8704e-30	2.6882e-30	7.0868e-30
	SSA	1.1478e-55	2.6161e-55	8.7204e-57	4.7764e-56
	IGWO	0	0	0	0
f_2	GWO	2.4416e-17	4.1886e-16	1.0329e-16	9.4679e-17
	WOA	1.1527e-22	6.1338e-20	2.1244e-21	1.1185e-20
	SSA	4.6396e-111	4.5959e-29	1.695e-30	8.3909e-30
	IGWO	0	0	0	0
f_3	GWO	3.1621e-08	3.9815e-4	2.0729e-05	67.4362e-05
	WOA	1.23e-08	0.00012959	1.2007e-05	2.865e-05
	SSA	6.4139e-180	1.0538e-24	3.5148e-26	1.9239e-25
	IGWO	0	0	0	0

Table 7. Comparative analysis of test results on multimodal functions
表 7. 多峰函数上的测试结果对比

Fun	Algorithm	Best	Worst	Mean	Std
f_4	GWO	0.310521	20.7612	2.8483	4.3607
	WOA	0	5.6843e-14	1.8948e-15	1.0378e-14
	SSA	6.7587e-08	1.1369e-13	3.7896e-15	2.0756e-14
	IGWO	0	0	0	0
f_5	GWO	7.1942e-14	1.6431e-13	1.0392e-13	1.9923e-14
	WOA	8.8818e-16	1.5099e-14	4.4409e-15	3.4909e-15
	SSA	8.8818e-16	8.8818e-16	8.8818e-16	0
	IGWO	0	0	0	0
f_6	GWO	6.839e-04	1.5721e-03	1.0392e-03	2.0883e-03
	WOA	1.23e-08	1.2959e-04	1.2007e-05	2.865e-05
	SSA	0	0	0	0
	IGWO	0	0	0	0

根据表 6 可以得出 IGWO 算法在单峰值函数上的优化性能较其他算法更强, IGWO 算法能够找到理论最优解。从表 7 中可以看出 IGWO 算法在多峰值测试函数上表现出较好地寻优性, 除了 f_6 , 其他函数上均能找到理论最优值。一般来说, 平均适应度越接近理论最小值, 算法的收敛精度越高; 而标准差越小, 算法的稳定性和鲁棒性越好。IGWO 算法在平均适应度和标准差方面的结果优于其他优化算法。这些结果表明, 综合使用四种改进策略的 IGWO 对算法寻优性能提升最为显著。

图 2 展示了不同算法在 6 个代表性基准函数上的收敛曲线。从图 2 中可以明显看出 IGWO 的收敛速度是所有算法中最快的。特别是对于多峰值函数, IGWO 算法表现出了更快的收敛速度, 在不到 50 次迭代的情况下收敛了。

3. 改进灰狼优化算法的图像分割

3.1. IGWO-Kmeans 图像分割

Kmeans 聚类算法是一种常用的基于距离的无监督学习方法, 广泛应用于图像分割领域[21]。该算法的原理是将图像中的像素根据它们的特征(如颜色或灰度)聚类成 k 个类别, 从而实现图像分割。

初始聚类中心的选取对于 Kmeans 算法的聚类效果具有重要影响, 而随机选择聚类中心的方式可能会导致聚类结果不理想, 甚至无法收敛。同时, Kmeans 算法在图像分割应用中聚类中心的选取直接影响图像分割的效果。而 GWO 具有良好的寻优能力, 帮助确定良好的初始聚类中心, 从而提升图像分割的效果。因此, 本文结合改进的 GWO 算法和 Kmeans 算法, 通过 IGWO 来最小化聚类中心和数据点之间距离的目标函数(14), 可以获得“最佳”聚类中心, 改善 Kmeans 算法的稳定性和性能, 从而提高图像分割的质量。即(14)作为 IGWO 的目标函数:

$$f = \sum_{i=1}^N \sum_{j=1}^K \|X_i - Y_j\|^2 \quad (14)$$

式中: X_i 为图像的第 i 个的灰度值, Y_j 为第 j 个聚类中心。

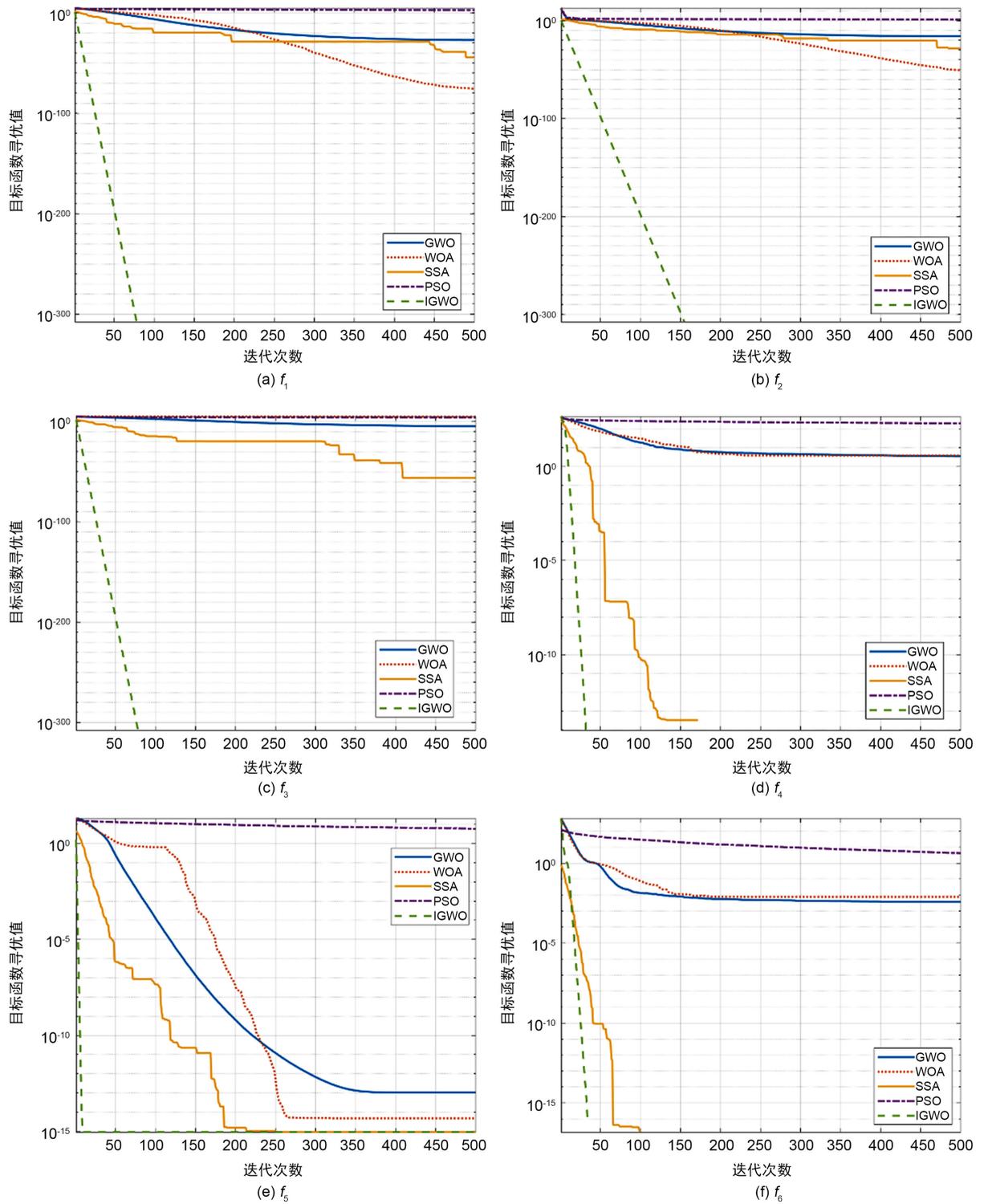


Figure 2. Function curve convergence diagram

图 2. 函数曲线收敛图

IGWO-Kmeans 聚类算法的步骤如下:

Step1 定义初始化参数, 确定分类数 k 和 IGWO 的基本参数;

- Step2 通过 IGWO 的全局搜索能力对初始聚类中心进行优化, 找到最佳的初始聚类中心;
 Step3 将 IGWO 找到的聚类中心作为 Kmeans 聚类算法的初始聚类中心;
 Step4 计算聚类平方误差和;
 Step5 更新聚类中心;
 Step6 对图像像素值进行聚类;
 Step7 判断是否达到终止条件, 如果没有则重复 Step4, Step5;
 Step8 输出聚类中心以及分割图。

3.2. 评价指标

单纯从主观视觉有时无法判断图像分割的细节差异, 为了评估所提出的方法在图像分割中的性能, 避免结果的随机性, 利用 PSNR、SSIM 和 FSIM 三个指标来分析分割后的图像。

PSNR: 衡量分割图像的质量, 具体算式如下[22]:

$$\text{PSNR} = 20 \lg \frac{255}{\text{RMSE}} \quad (15)$$

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^M \sum_{j=1}^N \left((I(i, j) - \text{Seg}(i, j))^2 \right)}{M \times N}} \quad (16)$$

式中: RMSE 为均方误差; I, Seg 分别为原始图与分割图; $M \times N$ 为图像大小。

SSIM: 原始图像和分割图像之间的相似性度量, 它考虑了亮度、对比度和结构相似性等多种因素, 具体算式如下[23]:

$$\text{SSIM}(I, \text{Seg}) = \frac{(2\mu_I \mu_{\text{Seg}} + c_1)(2\sigma_{I, \text{Seg}} + c_2)}{(\mu_I^2 + \mu_{\text{Seg}}^2 + c_1)(\sigma_I^2 + \sigma_{\text{Seg}}^2 + c_2)} \quad (17)$$

式中: μ_I 与 μ_{Seg} 分别为 I 与 Seg 的平均值; μ_I^2 与 μ_{Seg}^2 分别为 I 与 Seg 的方差; $\sigma_{I, \text{Seg}}$ 为 I 与 Seg 的协方差; c_1 与 c_2 为维持稳定的常数, 在实验里取 $c_1 = c_2 = 6.5025$ 。SSIM 的最高值意味着更好的性能。

FSIM: 评价原始图像和分割图像之间的特征相似性来衡量图像质量, 具体算式如下[23]:

$$\text{FSIM} = \frac{\sum_{x \in \Omega} S_L(x) PC_m(x)}{\sum_{x \in \Omega} PC_m(x)} \quad (18)$$

其中

$$S_L(x) = S_{PC}(x) S_G(x) \quad (19)$$

$$S_{PC}(x) = \frac{2PC_1(x)PC_2(x) + T_1}{PC_1^2(x)PC_2^2(x) + T_1} \quad (20)$$

$$S_G(x) = \frac{2G_1(x)G_2(x) + T_2}{G_1^2(x)G_2^2(x) + T_2} \quad (21)$$

$$G = \sqrt{G_x^2 + G_y^2} \quad (22)$$

$$PC(x) = \frac{E(x)}{\varepsilon + \sum_m A_n(x)} \quad (23)$$

式中: Ω 表示图像的整个区域; $S_L(x)$ 表示相似得分, $PC_m(x)$ 表示图像的相位一致性; $S_G(x)$ 表示; G

表示图像的梯度幅值; $A_n(x)$ 是尺度 n 上的局部振幅; $E(x)$ 是尺度 n 上位置 x 的响应矢量幅度; T_1, T_2 为常数, 实验中选择 $T_1 = 0.85$, $T_2 = 160$ 。FSIM 值越高, 预测性能越好。

3.3. 主观测评

选择了 4 幅 MIR 图像进行图像分割实验, 以评估 IGWO 对 Kmeans 算法的优化效果。同时选择原始的 GWO 和 WOA 对 Kmeans 算法进行优化(即 GWO-Kmeans, WOA-Kmeans), 并进行图像分割。此外, 将以上优化算法得到的图像分割结果与传统的 Kmeans 算法进行比较, 以评估不同方法对图像分割的性能提升程度。

聚类数目设置为 2, 最大迭代次数设置为 100, 进行图像分割。分割结果如图 3 所示:

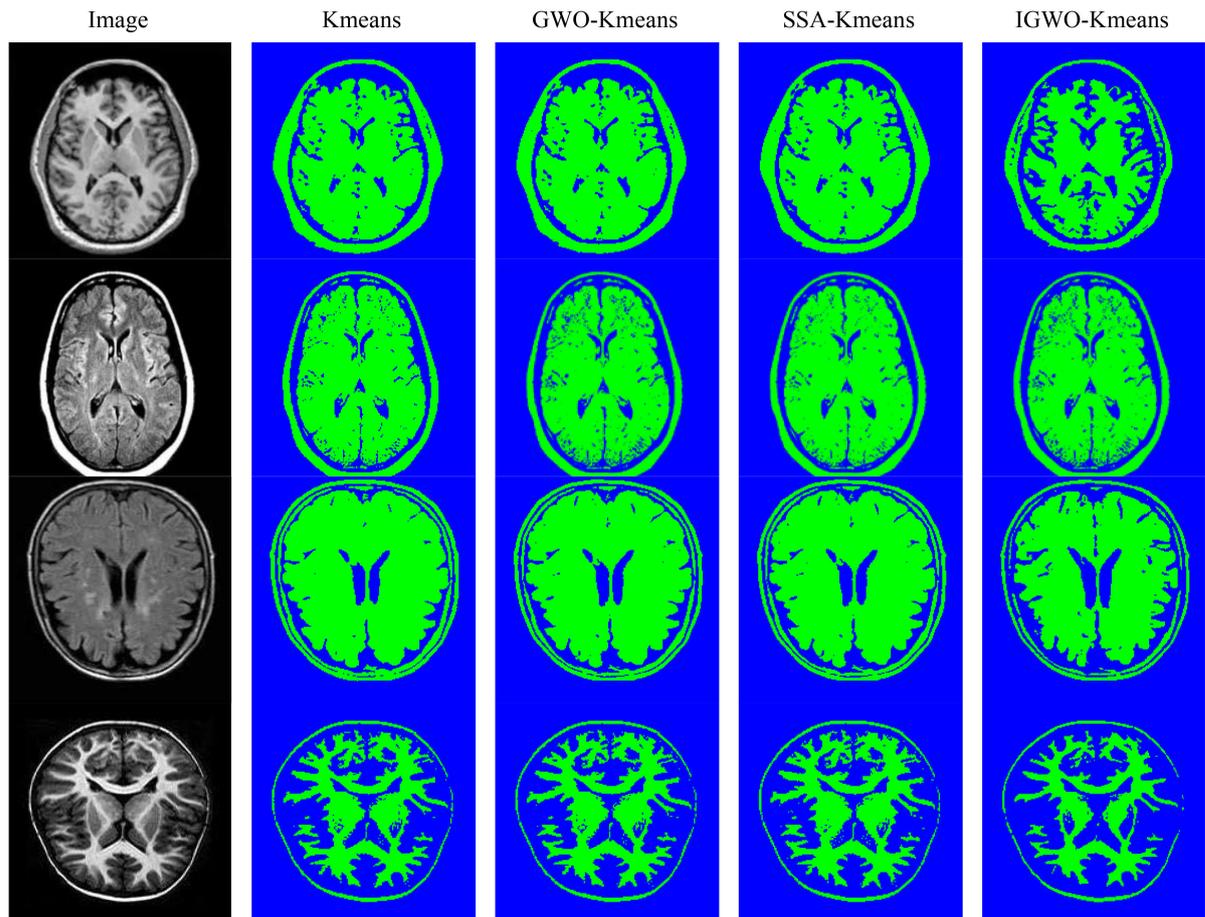


Figure 3. Comparison of Segmentation Results from Different Algorithms

图 3. 各算法分割结果对比

从图中可以看出, 传统的 Kmeans 算法在图像分割任务中表现最差, GWO-Kmeans 和 SSA-Kmeans 的分割效果相对接近。与之相比, IGWO-Kmeans 算法取得了更好的分割效果。这表明采用不同的优化算法能够显著改善 Kmeans 算法在图像分割中的性能。

3.4. 客观评测

表 8 给出了各算法的 PSNR, SSIM 和 FSIM 值。从表 8 中可知:

Table 8. Comparison of objective evaluation indicators for various algorithms
表 8. 各算法客观评价指标对比

Image	Th	Kmeans	GWO_Kmeans	SSA_Kmeans	IGWO_Kmeans
Image1	PSNR	47.4672	50.6255	50.7019	54.8024
	SSIM	0.8461	0.9367	0.9348	0.9710
	FSIM	0.8879	0.9456	0.9574	0.9661
Image2	PSNR	49.6232	50.4530	54.8963	50.3691
	SSIM	0.8383	0.9486	0.9324	0.9725
	FSIM	0.8906	0.9525	0.9646	0.9653
Image3	PSNR	48.6209	51.2975	51.5929	53.5768
	SSIM	0.8621	0.9459	0.9662	0.9651
	FSIM	0.8697	0.9552	0.9523	0.9623
Image4	PSNR	47.3548	49.8529	51.8586	55.4496
	SSIM	0.8822	0.9402	0.9228	0.9755
	FSIM	0.8304	0.9457	0.9614	0.9728

针对 Image1, IGWO-Kmeans 算法显示出了最佳的性能, 其 PSNR 值为 54.8024, 远超其他算法; 同时, IGWO-Kmeans 算法在 SSIM 和 FSIM 评价指标上也取得了最高分, 分别为 0.9710 和 0.9661。相比之下, 传统的 Kmeans 算法在所有指标上表现最差; 对于 Image2, IGWO-Kmeans 算法在 PSNR 指标上与 GWO-Kmeans 算法表现相近, 但在 SSIM 和 FSIM 上明显优于其他算法; 在 Image3 上, IGWO-Kmeans 在 PSNR 和 SSIM 上优于其他算法, 而在 FSIM 上与 GWO-Kmeans 并列第一; 针对 Image4, IGWO-Kmeans 算法在所有指标上均表现出色, 特别是在 FSIM 指标上取得了最高分。

综上所述, IGWO-Kmeans 算法在所有图像和所有评价指标下均表现出色, 特别是在 SSIM 和 FSIM 指标下取得显著优势。传统的 Kmeans 算法的性能明显落后于优化算法。

4. 结论

针对传统灰狼优化算法的不足, 为了进一步提升灰狼优化算法的寻优能力, 本文提出了一种多融合策略改进的灰狼优化算法(IGWO)。首先, 采用 Tent 混沌映射初始化种群, 从而增加种群多样性; 将线性收敛因子改为非线性收敛因子, 从而平衡 GWO 算法的探索能力和开发能力。其次, 利用透镜成像学习策略有效抑制 GWO 的过早收敛, 帮助种群跳出局部最优。最后, 引入 PSO 的思想改进 GWO 的位置更新方程, 增强灰狼个体优势, 避免陷入局部最优解。通过不同基准测试函数验证了改进算法 IGWO 的有效性。其次, 利用 IGWO 来获取初始最佳聚类中心, 并将其应用于 Kmeans 图像分割任务。对比传统的图像分割方法和其他优化算法的图像分割任务, 该方法在多个评价指标下表现出色, 获得最佳的图像分割效果。

基金项目

国家自然科学基金应急项目(61751316)和自治区自然科学特培项目(2022D03029)。

参考文献

- [1] Wu, M., Ye, H.-L., Wu, Y., et al. (2022) Brain Tumor Image Segmentation Based on Grouped Convolution. *Journal of*

- Physics: Conference Series*, **2278**, Article ID: 012042. <https://doi.org/10.1088/1742-6596/2278/1/012042>
- [2] Gutiérrez-Zaballa, J., Basterretxea, K., Echanobe, J., *et al.* (2023) On-Chip Hyperspectral Image Segmentation with Fully Convolutional Networks for Scene Understanding in Autonomous Driving. *Journal of Systems Architecture*, **139**, Article ID: 102878. <https://doi.org/10.1016/j.sysarc.2023.102878>
- [3] 卢才武, 宋义良, 江松, 等. 基于改进 U-net 的少样本煤岩界面图像分割方法[J]. *金属矿山*, 2024(1): 149-157.
- [4] 夏月月, 张以文. 一种融合三支决策理论的改进 K-means 算法[J]. *小型微型计算机系统*, 2020, 41(4): 724-731.
- [5] Lahbib, K., el Akkad, N., Satori, H., *et al.* (2022) A Performant Clustering Approach Based on an Improved Sine Cosine Algorithm. *International Journal of Computing*, **21**, 159-168. <https://doi.org/10.47839/ijc.21.2.2584>
- [6] Li, H., He, H. and Wen, Y. (2015) Dynamic Particle Swarm Optimization and K-Means Clustering Algorithm for Image Segmentation. *Optik*, **126**, 4817-4822. <https://doi.org/10.1016/j.ijleo.2015.09.127>
- [7] 董跃华, 李俊, 朱东林. 基于 Halton 序列改进蝠鲞算法的 K-means 图像分割[J]. *电光与控制*, 2023, 30(2): 91-98.
- [8] Mirjalili, S., Mirjalili, S.M. and Lewis, A. (2014) Grey Wolf Optimizer. *Advances in Engineering Software*, **69**, 46-61. <https://doi.org/10.1016/j.advengsoft.2013.12.007>
- [9] Wang, J.-S. and Li, S.-X. (2019) An Improved Grey Wolf Optimizer Based on Differential Evolution and Elimination Mechanism. *Scientific Reports*, **9**, Article No. 7181. <https://doi.org/10.1038/s41598-019-43546-3>
- [10] Gupta, S. and Deep, K. (2019) A Novel Random Walk Grey Wolf Optimizer. *Swarm and Evolutionary Computation*, **44**, 101-112. <https://doi.org/10.1016/j.swevo.2018.01.001>
- [11] Meidani, K., Hemmasian, A., Mirjalili, S., *et al.* (2022) Adaptive Grey Wolf Optimizer. *Neural Computing and Applications*, **34**, 7711-7731. <https://doi.org/10.1007/s00521-021-06885-9>
- [12] Fei, M.E.N. and Xi, J. (2020) Improved Gray Wolf Optimization Algorithm for Solving Low-Carbon Transportation Scheduling Problem in Open-Pit Mines. *Journal of Mine Automation*, **46**, 90-94.
- [13] Wang, Y., Zhang, X., Yu, D.-J., *et al.* (2022) Tent Chaotic Map and Population Classification Evolution Strategy-Based Dragonfly Algorithm for Global Optimization. *Mathematical Problems in Engineering*, **2022**, e2508414. <https://doi.org/10.1155/2022/2508414>
- [14] Long, W., Liang, X., Cai, S., *et al.* (2017) A Modified Augmented Lagrangian with Improved Grey Wolf Optimization to Constrained Optimization Problems. *Neural Computing and Applications*, **28**, 421-438. <https://doi.org/10.1007/s00521-016-2357-x>
- [15] Yang, J.C. and Long, W. (2016) Improved Grey Wolf Optimization Algorithm for Constrained Mechanical Design Problems. *Applied Mechanics and Materials*, **851**, 553-558. <https://doi.org/10.4028/www.scientific.net/AMM.851.553>
- [16] 龙文, 伍铁斌, 唐明珠, 等. 基于透镜成像学习策略的灰狼优化算法[J]. *自动化学报*, 2020, 46(10): 2148-2164.
- [17] Teng, Z., Lv, J. and Guo, L. (2019) An Improved Hybrid Grey Wolf Optimization Algorithm. *Soft Computing*, **23**, 6617-6631. <https://doi.org/10.1007/s00500-018-3310-y>
- [18] Mirjalili, S. and Lewis, A. (2016) The Whale Optimization Algorithm. *Advances in Engineering Software*, **95**, 51-67. <https://doi.org/10.1016/j.advengsoft.2016.01.008>
- [19] Xue, J. and Shen, B. (2020) A Novel Swarm Intelligence Optimization Approach: Sparrow Search Algorithm. *Systems Science & Control Engineering*, **8**, 22-34. <https://doi.org/10.1080/21642583.2019.1708830>
- [20] Gad, A.G. (2022) Particle Swarm Optimization Algorithm and Its Applications: A Systematic Review. *Archives of Computational Methods in Engineering*, **29**, 2531-2561. <https://doi.org/10.1007/s11831-021-09694-4>
- [21] Das, A., Namtirtha, A. and Dutta, A. (2023) Lévy-Cauchy Arithmetic Optimization Algorithm Combined with Rough K-Means for Image Segmentation. *Applied Soft Computing*, **140**, Article ID: 110268. <https://doi.org/10.1016/j.asoc.2023.110268>
- [22] Sharma, A., Chaturvedi, R. and Bhargava, A. (2022) A Novel Opposition Based Improved Firefly Algorithm for Multilevel Image Segmentation. *Multimedia Tools and Applications*, **81**, 15521-15544. <https://doi.org/10.1007/s11042-022-12303-6>
- [23] Peng, L. and Zhang, D. (2022) An Adaptive Lévy Flight Firefly Algorithm for Multilevel Image Thresholding Based on Rényi Entropy. *The Journal of Supercomputing*, **78**, 6875-6896. <https://doi.org/10.1007/s11227-021-04150-3>