

DOI:10.3969/j.issn.1673-4785.201410008

网络出版地址: <http://www.cnki.net/kcms/doi/10.3969/j.issn.1673-4785.201410008.html>

回溯搜索优化算法辅助的多阈值图像分割

尹雨山¹, 王李进^{1,2}, 尹义龙^{1,3}, 王冰清¹, 赵文婷¹, 徐云龙¹

(1. 山东大学 计算机科学与技术学院, 山东 济南 250101; 2. 福建农林大学 计算机与信息学院, 福建 福州 350002; 3. 山东财经大学 计算机科学与技术学院, 山东 济南 250014)

摘要: 阈值法是一种简单且有效的图像分割技术。然而阈值求解的计算量随阈值的增加而呈指数级别增长, 这给多阈值图像分割带来巨大挑战。为了克服计算量过大问题, 本文视多阈值分割模型为优化问题, 分别将 Otsu 法和 Kapur 法作为目标函数, 采用回溯搜索优化算法求解目标函数, 实现多阈值图像分割。将提出的多阈值分割算法应用于自然图像分割, 并与其他算法比较, 实验结果说明基于回溯搜索优化算法的多阈值图像分割技术是可行的, 而且具有较好的分割效果。

关键词: 阈值法; 回溯搜索优化算法; 图像分割; Otsu; Kapur; PSNR

中图分类号: TP183 **文献标志码:** A **文章编号:** 1673-4785(2015)01--

中文引用格式: 尹雨山, 王李进, 尹义龙, 等. 回溯搜索优化算法辅助的多阈值图像分割[J]. 智能系统学报, 2014, 10(1): -.

英文引用格式: YIN Yushan, WANG Lijin, YIN Yilong, et al. Backtracking search optimization algorithm assisted multilevel threshold for image segmentation[J]. CAAI Transactions on Intelligent Systems, 2014, 10(1): -.

Backtracking search optimization algorithm assisted multilevel threshold for image segmentation

YIN Yushan¹, WANG Lijin^{1,2}, YIN Yilong^{2,3}, WANG Binqing¹, ZHAO Wenting¹, XU Yunlong¹

(1. School of Computer Science and Technology, Shandong University, Jinan 250101, China; 2. College of Computer and Information Science, Fujian Agriculture and Forestry University, Fuzhou 350001, China; 3. School of Computer Science and Technology, Shandong University of Finance and Economics, Jinan 250014, China)

Abstract: Threshold method is a simple and effective image segmentation technique. Unfortunately, the calculated amount for solving threshold will be exponential amplification with the increase of threshold, resulting in a challenge problem for image segmentation. To deal with this problem, regarding image segmentation as optimization problem, we employ Otsu and Kapur methods to build two target functions, and then use backtracking search optimization algorithm to solve them, leading to segment image. The proposed approach is applied to nature images segmentation in comparison with other multilevel threshold optimized other algorithms. The results show it is feasible to segment image by using backtracking search optimization algorithm based multilevel threshold. Moreover, compared with other algorithms, it is suggested that the proposed approach can bring better performance.

Keywords: threshold method; backtracking search optimization algorithm; image segmentation; Otsu; Kapur; PSNR

图像分割就是指把图像分成各具特性的区域并

提取感兴趣目标的过程, 是图像处理到图像分析的关键步骤^[1], 也是计算机视觉的一个基本问题^[2]。图像分割多年来一直得到人们的高度重视, 至今已提出了各种各样的分割算法, 如基于阈值的分割方法^[3-7]、基于边缘检测的分割方法^[8-9]、基于区域的

收稿日期: 2014-10-08. 网络出版日期: .

基金项目: 国家自然科学基金-广东联合基金重点资助项目 (U1201258); 山东省自然科学基金杰出青年基金资助项目 (JQ201316).

通信作者: 王李进. E-mail: lijnawang@fafu.edu.cn.

分割方法^[10-12]、基于图论的分割方法^[13-14]、基于能量泛函的分割方法^[15-16]以及基于机器学习的分割方法^[17-20]等。

基于阈值的分割方法是各类分割算法中简单且广泛采用的方法,其基本思想是用一个或者多个阈值将待分割的图像的灰度级分为多个部分,灰度值在同一类中的像素属于同一个目标^[2]。因此,阈值的选取非常关键,并决定分割的结果。常见的计算阈值的方法主要有最大类间方差法(Otsu 算法)^[3]、最大熵法^[4-5]以及最小误差法^[7]等。上述计算阈值方法基本是在满足一定准则下通过解析式求得阈值,例如 Otsu 算法以目标和背景类间方差最大或类内方差最小为准则选取阈值。然而,通过解析式求解阈值的计算量和计算复杂度会随着阈值的增加而呈指数增长。因此,一些学者将基于准则函数的阈值求解问题视为以准则函数为目标函数的优化问题,于是出现了一些基于遗传算法^[21]、粒子群优化算法^[22]以及差分算法^[23]等的多阈值方法。得益于经典进化算法能有效求解多阈值问题,一些新颖的仿生算法用于该类问题,并呈现出较好的分割效果^[24-27]。

回溯搜索优化算法(backtracking search optimization algorithm, BSA)是一种新兴的仿生算法,其具有简单的结构,并能有效且快速求解各类函数优化问题^[28]。然而,关于 BSA 算法的应用研究报道较少,特别是在图像处理及应用领域。因此,借鉴于仿生算法求解多阈值问题的有效性,本文将 BSA 算法应用于图像分割,提出基于 BSA 算法的多阈值图像分割。提出的方法将 Otsu 算法和最大熵法的准则函数视为目标函数,并采用 BSA 算法分别获取多阈值,实现图像分割。实验说明提出的方法具有更好的性能。

1 阈值法

1.1 最大类间方差法(Otsu 法)

最大类间方差法给予判别分析最小二乘法的原理,其根据图像的灰度特性,将图像分为不同类别,各类之间方差要求最大。假设存在 m 级灰度的图像 P , 阈值 q 将图像的灰度值范围 $[0, 1, \dots, m-1]$ 分为背景与目标 2 部分。又设 p_i 表示灰度值为 i 出现的概率,则目标部分和背景的概率分别表示为

$$w_1 = \sum_{i=0}^q p_i \quad (1)$$

$$w_2 = \sum_{i=q+1}^{m-1} p_i \quad (2)$$

设 λ 、 λ_1 、 λ_2 分别表示图像、目标和背景的灰度值均值,则可表示为

$$\lambda_1 = \sum_{i=0}^q ip_i/w_1 \quad (3)$$

$$\lambda_2 = \sum_{i=q+1}^{m-1} ip_i/w_2 \quad (4)$$

且满足 $\lambda = w_1\lambda_1 + w_2\lambda_2$ 和 $w_1 + w_2 = 1$ 。

类间方差可表示为

$$d(q) = w_1(\lambda_1 - \lambda)^2 + w_2(\lambda_2 - \lambda)^2 \quad (5)$$

根据类间最大化准则,当方差取得最大值时,便得到最佳阈值 q 。

假设图像 P 存在 a 个阈值(q_1, q_2, \dots, q_a),式(5)容易扩展多阈值类间方差,可表示为

$$d(q_1, \dots, q_a) = w_1(\lambda_1 - \lambda)^2 + \dots + w_a(\lambda_a - \lambda)^2 \quad (6)$$

根据类间最大化准则,可通过计算式(7)获得最佳阈值:

$$(q_1, q_2, \dots, q_a) = \operatorname{argmax}(d(q_1, q_2, \dots, q_a))$$

1.2 最大熵法(Kapur 法)

20 世纪 80 年代以来,Shannon 信息熵的概念被应用于图像阈值化处理中,其思想是利用图像的灰度分布密度函数定义图像的信息熵,并根据优化准则求得阈值。文献[4]通过使后验的上限最大化准则确定阈值,而文献[5]假定目标和背景服从 2 个不同的概率分布,使得信息熵最大化求得最佳阈值。

假设存在 m 级灰度的图像 P , 阈值 q 将图像的灰度值范围 $[0, 1, \dots, m-1]$ 分为背景与目标 2 部分。又设 p_i 表示灰度值为 i 出现的概率,则目标和背景表示为式(1)和式(2),而它们的信息熵则可表示为

$$H_1 = - \sum_{i=0}^q \frac{p_i}{w_1} \ln\left(\frac{p_i}{w_1}\right) \quad (8)$$

$$H_2 = - \sum_{i=q+1}^{m-1} \frac{p_i}{w_2} \ln\left(\frac{p_i}{w_2}\right) \quad (9)$$

Kapur 方法^[5]是在图像 P 的总信息熵最大时,获得最佳阈值,即

$$q = \operatorname{argmax}(H_1 + H_2) \quad (10)$$

同样,式(10)很容易扩展为多阈值最大熵,可表示为

$$(q_1, q_2, \dots, q_a) = \operatorname{argmax}(H_1 + H_2 + \dots + H_a) \quad (11)$$

式中: a 表示阈值数目。

2 回溯搜索优化算法

BSA 算法是一种新兴的随机优化搜索技术,其结构简单,并且能够有效求解各类优化问题。另外,BSA 算法也是基于种群的搜索技术,并且使用一个外部文档维护其历史种群信息以引导种群进化。

当 BSA 算法用于求解优化问题时,首先在解搜索空间 $[x_{j, \min}, x_{j, \max}]$ ($j=1, 2, \dots, D$)内,通过均匀采

样初始化候选解 X 和历史种群 X_{old} :

$$x_{i,j,0} = x_{i,j,\min} + r(x_{i,j,\max} - x_{i,j,\min}) \quad i = 1, 2, \dots, NP \quad (12)$$

式中: $r \in [0, 1]$ 是随机数, NP 是种群大小。

与其他进化算法类似, BSA 算法使用 3 个基本的遗传操作: 变异、交叉和选择。

BSA 算法采用随机变异策略为每个个体生成中间候选个体 V_m 。该策略能够有效利用历史种群的信息引导算法进化, 具体公式为

$$V_m = X + F(X_{old} - X) \quad (13)$$

式中: F 缩放系数用以控制搜索方向矩阵。

其次, BSA 算法在变异个体 V_m 和当前种群 X 的基础上采用非均匀且较复杂的交叉策略生成候选解 T 。该策略通过随机方式生成一个映射矩阵 map ($NP \times D$), 并根据该矩阵将 V_m 和 X 中的信息映射成 T 。根据文献[28], 交叉策略可概括如算法 1 所示。

算法 1 交叉策略。

输入 变异个体 V_m 、种群 X 、种群规模 NP 、问题维数 D 、以及混合率 $mixrate$ 。

输出 候选解 T

- 1) 初始化矩阵 $map_{(1; NP, 1; D)} = 1$;
- 2) 均匀产生 2 个 $[0, 1]$ 之间的随机数 a 和 b ;
- 3) if $a > b$, 转入 4), 否则转入 5);
- 4) 进行如下操作后转入第 6 步:
 - for $i = 1$ to NP
 - 随机生成系列 $u = \text{permuting}(1; D)$;
 - 均匀生成 1 个 $[0, 1]$ 之间的随机数 c ;
 - 处理 $map_{(i, 1; u(1; mixrate \times c \times D))} = 0$;
 - end for
- 5) 进行如下操作:
 - for $i = 1$ to NP
 - 均匀生成 1 个 $[0, D]$ 之间的随机整数 d ;
 - 处理 $map_{(i, d)} = 0$;
 - end for
- 6) $T = V_m$;
- 7) 进行如下操作:
 - for $i = 1$ to NP
 - for $j = 1$ to D
 - if $map_{(i, j)} = 1$ then $T_{(i, j)} = P_{(i, j)}$;
 - end for
 - end for

另外, BSA 算法采用 2 种选择操作。第一种选择操作用于更新历史种群的信息, 其完全随机下接收当前种群信息, 可概括为

$$\text{if } a > b \text{ then } X_{old} = X \quad a, b \sim U(0, 1) \quad (14)$$

第 2 种选择操作则根据当前种群 X 和候选种群 T 的适应值, 贪婪选择适应值较好的个体进入下

一代。

3 应用 BSA 求解多阈值

应用 BSA 算法求解多阈值问题, 其实质是将多阈值准则作为目标函数, 采用 BSA 算法搜索最优阈值, 具体步骤如算法 2 所示。

算法 2 基于 BSA 算法的多阈值图像分割。

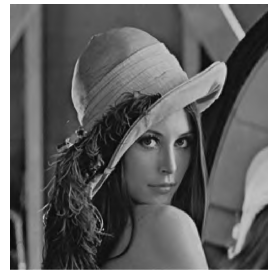
输入 种群规模 NP 、问题维数 D (阈值数目)、混合率 $mixrate$ 、最大迭代次数 $MaxIteration$ 。

输出 最佳阈值 q

- 1) 采用式 (12) 初始化种群 X 和历史种群 X_{old} ;
- 2) 初始化迭代计数器 $iter = 1$;
- 3) if $iter > MaxIteration$, 转入 11);
- 4) 执行第 1 种选择操作, 即执行式 (14) 更新历史种群;
- 5) 执行变异操作, 即执行式 (13);
- 6) 执行交叉操作获得 T , 即执行算法 1;
- 7) 采用式 (7) 或者式 (11) 评价 T ;
- 8) 根据 X 和 T 的适应值, 采用第 2 种选择操作获得下一代种群 X 。
- 9) 获得当前最优阈值 q ;
- 10) $iter = iter + 1$, 转入 3);
- 11) 输出最优阈值 q 。

4 实验与结果

为了分析 BSA 算法的多阈值图像分割性能, 本文采用文献[25]中的 Camera、Lena、Pepper 以及 Baboon 等 4 幅图像作为待分割图像见图 1, 其中, 每幅图像的大小为 256×256 。



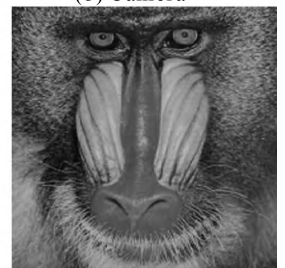
(a) Lena



(b) Camera



(c) Pepper



(d) Baboon

图 1 测试图像

Fig.1 Test images

另外,图像峰值信噪比(peak signal to noise ratio, PSNR)作为性能指标,其中 PSNR 公式^[24]如下:

$$PSNR = 20 \log_{10} \left(\frac{255}{RMSE} \right) \quad (15)$$

式中:

$$RMSE = \sqrt{\frac{1}{M \cdot N} \sum_{i=1}^M \sum_{j=1}^N (I(i,j) - J(i,j))^2} \quad (16)$$

式中:图像 I 大小为 $M \times N$, J 为阈值化后的图像。

在实验中,各算法针对每幅图像独立运行 30 次。每次独立运行中,最大迭代数 MaxIteration 为 160,种群大小 NP 为 20。

4.1 Otsu 方法的实验结果

表 1 给出了与基于传统优化算法的多阈值 Otsu (MOT)^[29] 比较的实验结果。MOT 中的适应值是将 MOT 中的阈值带入式(7)求得。表 2 列出与基于细菌算法(bacterial foraging algorithm, BFA)的 Otsu 多阈值^[25]和带惯性权重 PSO 算法^[30]的 Otsu 多阈值的实验结果。其中,PSO 算法的最大和最小惯性权重分别为 0.9 和 0.4;BFA 算的参数见文献[25]。另外,图 2 给出各算法求解的 PSNR 随 Otsu 阈值数的变化趋势。

表 1 MOT 和 BSA 算法的 Otsu 多阈值目标函数适应值和 PSNR

Table 1 Multi-threshold Otsu fitness and PSNR obtained by MOT and BSA

测试图像	阈值数目	MOT			BSA		
		阈值	适应值	PSNR	阈值	适应值	PSNR
Lena	2	85,143	2 895.36	12.67	85,156	2 923.59	13.65
	3	74,120,164	3 136.52	15.83	71,127,181	3 174.28	17.16
	4	66,105,137,173	3 238.31	18.01	63,111,150,192	3 271.09	19.38
	5	59,88,115,142,176	3 276.55	18.93	61,105,138,164,200	3 309.50	20.94
Camera	2	69,143	3 927.90	11.14	64,141	3 929.46	11.12
	3	59,121,157	4 010.71	12.72	54,118,155	4 012.30	12.85
	4	59,116,148,173	4 053.73	16.13	36,90,136,167	4 069.13	16.38
	5	45,97,135,162,196	4 092.03	19.87	31,81,122,149,172	4 103.01	19.17
Pepper	2	65,132	3 142.25	12.505 3	71,147	3 180.47	12.87
	3	61,116,164	3 364.89	15.16	66,129,183	3 396.53	16.57
	4	57,100,136,171	3 445.22	16.28	46,90,135,185	3 476.78	17.69
	5	44,79,110,143,173	3 498.36	17.04	42,83,121,158,193	3 533.00	19.47
Baboon	2	99,145	1 566.10	12.49	119,177	1 742.75	15.85
	3	86,121,154	1 722.19	13.93	101,146,188	1 865.12	18.22
	4	73,106,132,158	1 782.34	14.64	86,127,162,196	1 934.25	19.69
	5	42,81,110,133,161	1 821.35	15.22	85,120,146,173,203	1 967.98	21.53

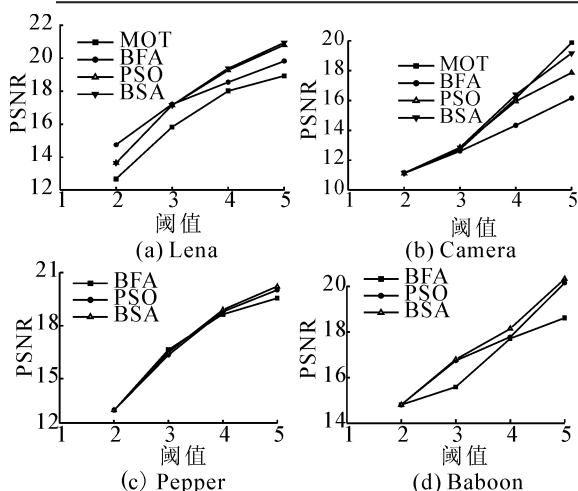


图 2 T-S 逻辑的基本框图

Fig.2 Basic block diagram of T-S logical

从表 1 可知,采用 BSA 算法求解的阈值使得适

应值都优于 MOT 的所得适应值;另外借助于 PSNR, BSA 算法也优于 MOT。上述结果说明了 BSA 算法以 Otsu 的最大类间准则为目标函数求解多阈值是可行的,而且获得较好的性能。

从表 2 可以看出,与 BFA 算法比较,BSA 算法求解的多阈值在目标函数适应值以及 PSNR 上都明显较优。另外,与 PSO 算法比较,在各测试图像的 2 和 3 个阈值上,BSA 算法求解的多阈值与 PSO 算法求解的多阈值是相同的,然而在 4 和 5 个阈值上,PSO 算法获得稍微较好的目标函数适应值,但是 BSA 算法却获得较好的 PSNR。总体而言,BSA 算法的多阈值与带惯性权重 PSO 算法的多阈值性能是相同的。图 2 同样说明了随阈值数的增加,BSA 算法求解的 PSNR 趋势总体上是最好的。

表2 BFA 算法、PSO 算法和 BSA 算法求解的 Otsu 多阈值目标函数适应值和 PSNR
Table 2 Multi-threshold Otsu fitness and PSNR obtained by BFA, PSO and BSA

测试 图像	阈值 数目	BFA			PSO			BSA		
		阈值	适应值	PSNR	阈值	适应值	PSNR	阈值	适应值	PSNR
Lena	2	84,155	2923.58	14.75	85,156	2923.59	13.65	85,156	2923.59	13.65
	3	78,138,186	3165.87	17.21	71,127,181	3174.28	17.16	71,127,181	3174.28	17.16
	4	62,105,133,190	3243.88	18.55	63,110,150,192	3271.09	19.27	63,111,150,192	3271.09	19.38
	5	60,81,107,140,181	3281.39	19.83	62,106,138,165,200	3309.50	20.81	61,105,138,164,200	3309.50	20.94
Camera	2	64,141	3929.46	11.12	64,141	3929.46	11.12	64,141	3929.46	11.12
	3	56,115,152	4010.73	12.60	54,118,155	4012.30	12.85	54,118,155	4012.30	12.85
	4	22,87,124,161	4056.59	14.33	36,91,137,168	4069.16	15.94	36,90,136,167	4069.13	16.38
	5	40,75,106,142,166	4090.59	16.16	29,79,121,149,173	4103.20	17.86	31,81,122,149,172	4103.01	19.17
Pepper	2	73,147	3180.16	12.79	71,147	3180.47	12.87	71,147	3180.47	12.87
	3	69,127,183	3394.44	15.25	66,129,183	3396.53	16.57	66,129,183	3396.53	16.57
	4	40,94,125,177	3454.18	16.55	47,91,136,185	3476.79	17.68	46,90,135,185	3476.78	17.69
	5	50,93,144,169,193	3506.14	17.78	43,84,122,159,194	3533.05	19.20	42,83,121,158,193	3533.00	19.47
Baboon	2	118,176	1742.52	15.10	119,177	1742.75	15.85	119,177	1742.75	15.85
	3	107,154,199	1856.36	16.97	101,146,188	1865.12	18.22	101,146,188	1865.12	18.22
	4	73,132,168,208	1905.70	19.59	88,127,162,197	1934.27	19.69	86,127,162,196	1934.25	19.69
	5	79,110,133,166,200	1961.01	21.20	82,118,147,172,202	1968.39	21.25	85,120,146,173,203	1967.98	21.53

表3 BFA 算法、PSO 算法和 BSA 算法求解的 Kapur 多阈值目标函数适应值和 PSNR
Table 3 Multi-threshold Kapur fitness and PSNR obtained by BFA, PSO and BSA

测试 图像	阈值 数目	BFA			PSO			BSA		
		阈值	适应值	PSNR	阈值	适应值	PSNR	阈值	适应值	PSNR
Lena	2	92,173	17.72	14.62	92,173	17.72	14.62	92,173	17.72	14.62
	3	66,127,191	21.95	17.02	72,127,188	21.98	17.03	72,126,188	21.98	17.10
	4	40,91,133,179	25.63	18.84	52,92,140,189	25.84	19.19	51,92,140,189	25.84	18.96
	5	47,83,126,186,211	29.22	19.72	48,87,128,172,207	29.52	20.46	48,87,129,172,208	29.52	20.71
Camera	2	125,192	17.56	11.66	124,192	17.56	12.08	124,192	17.56	12.08
	3	36,99,194	21.95	15.60	37,100,192	21.97	15.69	37,100,192	21.97	15.69
	4	57,100,143,196	26.29	19.13	37,92,143,196	26.54	18.99	37,92,143,196	26.54	18.99
	5	19,75,118,152,201	30.23	20.91	18,56,95,143,196	30.50	20.23	18,57,94,144,196	30.50	20.05
Pepper	2	81,161	18.14	13.22	81,161	18.14	13.22	81,161	18.14	13.22
	3	61,118,180	22.52	16.64	62,121,180	22.53	16.34	61,121,180	22.53	16.49
	4	64,111,154,209	26.55	18.64	61,113,163,214	26.66	18.79	62,113,163,214	26.66	18.89
	5	52,81,114,159,217	30.40	19.55	43,82,122,166,214	30.59	20.02	43,82,124,169,214	30.59	20.21
Baboon	2	93,170	17.22	14.81	93,170	17.22	14.81	93,170	17.22	14.81
	3	83,133,194	21.42	15.59	71,127,184	21.48	16.75	71,126,184	21.48	16.80
	4	48,91,128,191	25.35	17.71	48,93,138,189	25.44	17.78	48,93,138,190	25.44	18.15
	5	44,79,132,164,204	28.99	18.62	48,88,128,168,205	29.13	20.16	46,88,130,169,205	29.13	20.34

4.2 Kapur 方法的实验结果

表3 给出了不同仿生算法求解 Kapur 多阈值的比较结果,其中参数与 4.1 节相同。

从表3 可以看出,BSA 算法求解的目标函数适应值上完全优于 BFA 算法求解的目标函数适应值,而且借助于 PSNR 性能,BSA 算法也优于 BFA 算法。另外与 PSO 算法比较,BSA 算法求解的目标函数适

应值基本上相似,但借助于 PSNR,BSA 算法的多阈值法总体上优于 PSO 算法的多阈值法。

图3 给出各仿生算法求解的 PSNR 随 Kapur 阈值数的变化趋势。从图3 可以看出,在多数图像上,BSA 求解的 PSNR 随 Kapur 阈值数的变化趋势优于其他 2 种算法。

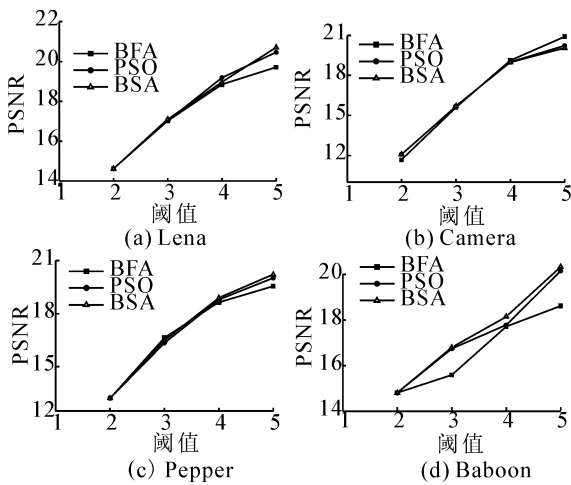


图 3 PSNR 随 Kapur 阈值变化的趋势

Fig.3 The trend of PSNR change with Kapur threshold

5 结束语

本文将 BSA 算法应用于图像分割,提出 BSA 算法求解的多阈值图像分割。提出方法将 Otsu 方法和 Kapur 方法的求多阈值准则函数作为目标函数,应用 BSA 算法求解,并实现图像分割。仿真结果说明 BSA 算法求解的多阈值图像分割是可行的,与其他 BFA 算法和 PSO 算法求解的多阈值分割方法比较,本文提出的方法具有较好的性能。下一步工作将提出方法应用于更多的图像测试,包括遥感图像以及医学影像等。

参考文献:

[1] 章毓晋. 图像工程 [M]. 北京:清华大学出版社, 2002: 179-186.

[2] 刘国英, 马国锐, 王雷光, 等. 基于 Markov 随机场的小波域图像建模及分割—Matlab 环境 [M]. 北京:科学出版社, 2010:6-15.

[3] OTSU N. A threshold selection method from gray-level histograms[J]. Automatica, 1975, 11: 23-27.

[4] PUN T. A new method for grey-level picture thresholding using the entropy of the histogram [J]. Signal Processing, 1980, 2(3): 223-237.

[5] KAPUR J N, SAHOO P K, WONG A K C. A new method for gray-level picture thresholding using the entropy of the histogram[J]. Computer Vision, Graphics and Image Processing, 1985, 29(3): 273-285.

[6] REDDI S S, RUDIN S F, KESHAVAN H R. An optimal multiple threshold scheme for image segmentation[J]. IEEE Transactions on Systems, Man, and Cybernetics, 1984, 14(4): 661-665.

[7] KITTLER J, ILLINGWORTH J. Minimum error thresholding [J]. Pattern Recognition, 1986, 19(1): 41-47.

[8] CANNY J. A computational approach to edge detection[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1986, PAMI-8(6): 679-698.

[9] ZIOU D, TABBONE S. Edge detection techniques: An overview[J]. International Journal of Pattern Recognition and Image Analysis, 1998, 8(4): 537-559.

[10] CHEN PC, PAVLIDIS T. Segmentation by texture using a co-occurrence matrix and a split-and-merge algorithm[J]. Computer Graphics Image Processing, 1979, 10(2): 172-182.

[11] CHEN S Y, LIN W C, CHEN C T. Split-and-merge image segmentation based on localized feature analysis and statistical tests[J]. CVGIP: Graphical Models and Image Processing, 1991, 53(5): 457-475.

[12] CHANG Y L, LI X. Adaptive image region-growing [J]. IEEE Transactions On Image Processing, 1994, 3(6): 868-872.

[13] BOYKOV Y, JOLLY MP. Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images [C]//Proceedings of the Eighth International Conference on Computer Vision. Piscataway, NJ: IEEE, 2001: 105-112.

[14] GRADY L. Random walks for image segmentation [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2006, 28(11): 1768-1783.

[15] KASS M, WITKIN A, TERZOPOULOS D. Snakes: active contour models [J]. International Journal of Computer Vision, 1988, 1(4): 321-331.

[16] OSHER S, SETHIAN J A. Fronts propagating with curvature-dependent speed: algorithms based on Hamilton - Jacobi formulations [J]. Journal of Computational Physics, 1988, 79(1): 12-49.

[17] ROUT S, SRIVASTAVA M J. Multimodal image segmentation using a modified hopfield neural network [J]. Pattern Recognition, 1998, 31(6): 743-750.

[18] 林开颜, 徐立鸿, 吴军辉. 快速模糊 C 均值聚类彩色图像分割方法 [J]. 中国图像图像学报, 2004, 9(2): 159-163.

[19] CAO G B, WANG S L, WEI B Z, et al. A hybrid cnn-rf method for electron microscopy images segmentation [J]. Journal of Biomimetics Biomaterials and Tissue Engineering, 2013, 18(2): 1-6.

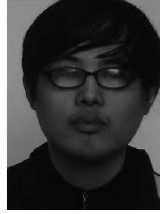
[20] WANG S L, CAO G B, WEI B Z, et al. Hierarchical level features based trainable segmentation for electron microscopy images [J]. Biomedical Engineering Online, 2013, 12

(1): 59-72.

- [21] TANG K Z, YUAN X J, SUN T K, et al. An improved scheme for minimum cross entropy threshold selection based on genetic algorithm[J]. Knowledge-Based Systems, 2011, 24(8): 1131-1138.
- [22] YIN P Y. Multilevel minimum cross entropy threshold selection based on particle swarm optimization[J]. Applied Mathematics and Computation, 2007, 182(2): 503-513.
- [23] ALI M, AHN C W, PANT M. Multi-level image thresholding by synergetic differential evolution[J]. Applied Soft Computing, 2014, 17:1-11.
- [24] AGRAWAL S, PANDA R, BHUYAN S, et al. Tsallis entropy based optimal multilevel thresholding using cuckoo search algorithm[J]. Swarm and Evolutionary Computation, 2013, 11: 16-30.
- [25] SATHYA P D, KAYALVIZHI R. Optimal multilevel thresholding using bacterial foraging algorithm[J]. Expert System with Application, 2011, 38(12): 15549-15564.
- [26] HORNG M H, LIOU R J. Multilevel minimum cross entropy threshold selection based on the firefly algorithm[J]. Expert System with Application, 2011, 38(12): 14805-14811.
- [27] MA M, LIANG J H, GUO M, et al. SAR image segmentation based on artificial bee colony algorithm[J]. Applied Soft Computing, 2011, 11(8): 5205-5214.
- [28] CIVICIOGLU P. Backtracking search optimization algorithm for numerical optimization problems[J]. Applied Mathematics and Computation, 2013, 219: 8121-8144.
- [29] MATHWORKS. Multilevel image thresholds using Otsu's method [EB/OL]. [2014-08-22]. <http://www.mathworks.cn/cn/help/images/ref/multithresh.html>.

- [30] SHI Y H, EBERHART R C. A modified particle swarm optimizer[C]//Proceedings of the 1998 IEEE World Congress on Computational Intelligence. Piscataway, NJ: 1998:69-73.

作者简介:



尹雨山,男,1990年生,硕士研究生,主要研究方向为智能信息处理及其应用。



王李进,男,1977年生,副教授,主要研究方向为计算智能及其应用。



尹义龙,男,1972年生,教授,博士生导师,博士,主要研究方向为机器学习、数据挖掘、生物特征识别。