

局部颜色模型的交互式 Graph-Cut 分割算法

郑加明, 陈昭炯

(福州大学 数学与计算机科学学院, 福建 福州 350108)

摘要: 由于目前大多数交互式 Graph-Cut 分割算法很难达到精确分割且实时交互的效果. 对此, 提出一种基于局部颜色模型的改进算法. 该算法利用 Mean-Shift 预分割, 建立基于局部颜色模型的交互式分割框架, 并将像素级的 Graph-Cut 算法转化为基于区域的算法进行快速求解. 预分割之后的区域保持了原有图像的结构, 不仅提高了采用局部颜色模型估计分布的准确性, 而且基于区域 Graph-Cut 的算法明显降低了计算的复杂度. 实验结果表明, 改进后的算法不仅保证了分割的精确性, 而且还达到了实时交互.

关键词: Graph-Cut; 交互式图像分割; Mean-Shift; 实时交互性

中图分类号: TP391 **文献标识码:** A **文章编号:** 1673-4785(2011)04-0318-06

Interactive Graph-Cut for image segmentation using local color models

ZHENG Jiaming, CHEN Zhaojiong

(College of Mathematics and Computer Science, Fuzhou University, Fuzhou 350108, China)

Abstract: Most interactive segmentation algorithms based on Graph-Cut do not usually lend themselves to real time applications with accurate segmentation. In this paper, an improved algorithm using local color models was proposed to deal with the problem. Using Mean-Shift technology, the proposed algorithm built an interactive image segmentation framework based on local color models. A Graph-Cut algorithm was then applied to the pre-segmented regions instead of image pixels. The pre-segmented regions preserve the image structure, which improves the estimation accuracy of distribution based on local color models and dramatically reduces the computational complexity. Experimental results show that the proposed algorithm has good real time interactivity with accurate segmentation.

Keywords: Graph-Cut; interactive image segmentation; Mean-Shift; real time interactivity

交互式图像分割方法主要用于提取静态图像或视频序列中的前景目标, 其目的是希望通过尽量少的用户交互, 快速精确地提取出感兴趣的对象, 该方法在一定程度上解决了全自动分割存在的通用性差和分割效果不精确等问题.

近年来, 有关交互式图像分割的算法已越来越多, 如蛇形算法^[1]、交互式 Graph-Cut 分割算法^[2]、Grabcut^[3]和 Lazy snapping^[4]等. 其中, 交互式 Graph-Cut 分割算法将交互式图像分割转化为求解一个包含区域和边界信息能量函数的最小化过程, 很好地将图像的区域信息与边界信息结合起来, 分割的效果较为准确.

然而, 大多数交互式 Graph-Cut 分割算法还很难达到交互式分割所要求的快速精确分割的效果. 这主要由两方面原因造成的. 一方面, 由于图像的像素个数一般是海量的, 因而基于像素级的 Graph-Cut 算法的计算量和储存量过大. 为了解决此问题, 可以通过对图像进行预分割来降低算法的计算复杂度. 如文献[4]提出的 Lazy snapping 算法, 将图像进行分水岭过分割, 建立基于区域的 Graph-Cut 算法, 在一定程度上实现了实时交互. 但是采用分水岭过分割, 对原有图像的结构会造成破坏, 需要更多的交互以弥补过分割带来的错误, 另一方面, 采用全局颜色模型不能反映图像的空间结构, 对复杂图像分布的估计不准确. 如文献[3-6]提出的改进 Graph-Cut 算法, 都采用高斯混合模型 (Gaussian mixture model, GMM) 对前景和背景的颜色分布进行估计. 采用高

收稿日期: 2010-06-05.

基金项目: 国家自然科学基金资助项目 (60805042); 福建省自然科学基金资助项目 (2010J01329).

通信作者: 陈昭炯. E-mail: chenzz@fzu.edu.cn.

斯混合模型虽然比文献[2]采用直方图进行分布估计要准确,却耗费了大量的时间,影响分割的速度.而且这种采用全局颜色模型的算法,很难反映图像的空间信息,不能准确估计复杂图像的分布.

为了在保证分割效果精确的同时,提高交互式分割的效率,提出一种基于局部颜色模型的交互式 Graph-Cut 分割算法.新算法采用 Mean-Shift 对图像进行预分割,利用预分割之后的区域集来估计前景和背景分布的候选局部颜色模型,并建立基于区域的 Graph-Cut 模型.由于 Mean-Shift 能够很好地分析具有多峰分布的特征空间^[7],因此所建立的候选局部颜色模型能够较好地估计出图像的分布.又由于 Mean-Shift 分割考虑了图像的颜色特征和空间信息,分割后的区域能够很好地保持原有图像的结构,因此采用基于区域的 Graph-Cut 模型,可以大大减少每次交互的计算量.而且在每次迭代时,前景和背景分布的模型可根据用户标记直接选取候选模型来重新建立分布,进一步减少了交互的延迟时间.另外,还根据前景和背景局部颜色模型的重叠程度,计算估计分布可能造成的错误率,自适应地调整模型的参数.从实际检验的结果看,改进后的算法不仅保证了分割的精确性,还明显减少了交互延迟时间.

1 交互式 Graph-Cut 分割框架

1.1 交互式图像分割的能量函数

交互式图像分割问题实际上是一个二值标记组合优化问题.通常,将一幅图像看作一个无向图 $G = \langle V, E \rangle$,其中顶点集 V 对应所有的像素,边集 E 对应各个像素与相邻像素的邻接关系.二值标记组合优化问题的目标就是根据图 G ,按照一定原则,找到一个最优解 X ,解中每个顶点都标记上惟一的标签(背景为“back”,前景为“fore”).最优解 X 可以采用 $E(X)$ 能量函数求得:

$$E(X) = R(X) + \lambda B(X).$$

式中: $R(X)$ 是用来反应区域信息的区域项能量, $B(X)$ 是用来反应边界信息的边界项能量,区域项与边界项的比重由参数 λ 确定.

1.2 Graph-Cut 算法求最小化能量函数的过程

Graph-Cut 算法需要用户采用基于笔画的交互形式对前景和背景进行标记,被标记上的像素称为前景或背景种子,并根据前景和背景种子估计前景和背景的颜色分布.

区域项 $R(X)$ 用来惩罚解 X 与观察数据不一致的程度.一般可以将区域项定义为如下形式:

$$R(X) = \sum_{i \in V} R(x_i).$$

通常, $R(x_i)$ 用来反映像素 x_i 颜色特征适合前景或者背景颜色分布的程度.

边界项 $B(X)$ 用来刻画解 X 的边界信息,其定义形式如下:

$$B(X) = \sum_{(i,j) \in E} B(x_i, x_j) \delta(x_i, x_j).$$

式中: $\delta(x_i, x_j) = \begin{cases} 1, & x_i \neq x_j; \\ 0, & \text{otherwise.} \end{cases}$ $B(x_i, x_j)$ 用来衡量像素 x_i 与像素 x_j 的平滑程度.

为了求解能量函数 $E(X)$ 的最优解, Graph-Cut 算法在无向图 $G = \langle V, E \rangle$ 的基础上构建源点 s 和汇点 t 的 s - t 网络流图.源点、汇点与图 G 中的每个顶点由一条边相连,边的权重由 $R(x_i)$ 计算得到.图 G 中顶点邻接边的权重由 $B(x_i, x_j)$ 计算得到. Graph-Cut 算法利用最大流/最小割原理^[2],通过求 s - t 网络流图中的最大流,解得能量的最小值,即最小割.

2 改进的交互式 Graph-Cut 分割算法

2.1 基于局部颜色模型的能量函数构造

局部颜色模型融合了原有图像的空间结构与颜色信息,可以更好地估计复杂图像的分布.因此,采用局部颜色模型对前景和背景的分布进行估计,构造基于局部颜色模型的能量函数.

利用 Mean-Shift 预分割来估计前景和背景分布的候选局部颜色模型.先对图像中所有 n 个像素 $\{z_i\}_{i=1,2,\dots,n}$ 经过 Mean-Shift 过滤,再合并相似区域和小区域,得到 m 个聚簇.这 m 个聚簇的集合就是图像分布的 m 个模式,也就是候选局部颜色模型 $\{C_p\}_{p=1,2,\dots,m}$.对于每一个像素 z_i 所属的局部颜色模型为 $L_i = \{p | z_i \in C_p\}$,对应的模式特征值为 $y_i = \{M(p) | z_i \in C_p\}$.接着,根据用户标记从候选局部颜色模型 $\{C_p\}_{p=1,2,\dots,m}$ 中选取前景和背景颜色模型.设用户标记的 g 个前景种子集合为 $\{F_g\}$, h 个背景种子集合为 $\{B_h\}$.如果 $z_i \in \{F_g\}$,则候选局部颜色模型中的 L_i 模型晋升为前景颜色模型.若晋升 $\text{num}(F)$ 个局部颜色模型为前景颜色模型,将其定义为 θ_F .同样,可以得到 $\text{num}(B)$ 个局部颜色模型为背景颜色模型,定义为 θ_B .则像素 z_i 属于前景和背景分布的概率为 $P(z_i | \theta_F)$ 和 $P(z_i | \theta_B)$.当用户添加新的标记时,前景和背景分布的模型根据用户标记直接从候选模型中选取,不需要重新进行 Mean-Shift 分割,提高了算法的效率.

为了达到实时交互,将预分割后的区域作为图的顶点,并在此基础上构造基于局部颜色模型的能量函数.设基于区域的图结构为 $G' = \langle V', E' \rangle$,其

中顶点 p 的值为该区域的模式特征值 $M(p)$, 若区域中任一像素与另一区域中某像素满足 8-邻域关系, 则这 2 个区域对应的顶点是相邻关系. 本文所定义的种子是指包含前景或背景种子的区域块. 对于误分割产生的错误结果, 可以对局部区域进行基于像素的 Graph-Cut 分割. 对此, 将主要研究基于区域的交互式 Graph-Cut 分割算法, 构造基于局部颜色模型的能量函数如下:

$$E(X) = \mu \sum_{i \in V'} R(x_i) + \lambda \sum_{(i,j) \in E'} (x_i, x_j) \delta(x_i, x_j). \quad (1)$$

式中: μ 为自适应因子, 由前景与背景颜色模型估计分布产生的错误率 ρ 确定, 可定义为

$$\mu = \begin{cases} 1 - \rho, & \rho < \delta; \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

式中: δ 为常量因子, ρ 可根据贝叶斯分类的错误率定义为

$$\rho = \frac{1}{2} \left[\frac{\sum_{p \in \theta_F} P(M(p) | \theta_B)}{\text{num}(F)} + \frac{\sum_{p \in \theta_B} P(M(p) | \theta_F)}{\text{num}(B)} \right]. \quad (3)$$

2.2 利用最大流原理求解能量函数的最小值

将能量函数(1)中的区域项和边界项能量映射到 s - t 网络流图, 并将图 G' 中的顶点分为前景种子、背景种子和未确定区域 3 种类型. 其中, 未确定区域是指除了用户标记过的前景和背景种子外的其他区域. 顶点与源点 s 、汇点 t 连接的权重分别为 $R(x_{\text{fore}})$ 和 $R(x_{\text{back}})$. 不同类型顶点, 所对应的 $R(x_{\text{fore}})$ 和 $R(x_{\text{back}})$ 的值如表 1 所示. 为了保证种子确定属于前景或者背景, $L(i)$ 必须大于顶点 i 邻接边权重的总和. 根据顶点 i 的邻接边条数 $\text{neigh}(i)$ 不同, $L(i)$ 的计算方法如下:

$$L(i) = \lambda \text{neigh}(i) + 1.$$

当顶点 i 属于未确定区域, 根据前景和背景分布计算顶点 i 区域项的值 $S_{\text{fore}}(i)$ 和 $S_{\text{back}}(i)$. 另外将边界项能量定义为相邻顶点 x_i 和 x_j 的权值:

$$B(x_i, x_j) = \exp\left(-\frac{\|M(i) - M(j)\|^2}{\sigma^2}\right). \quad (4)$$

式中: σ 为所有邻接边权重的平均值.

表 1 区域项的值

Table 1 Region term values

顶点 i 类型	$R(x_{\text{fore}})$	$R(x_{\text{back}})$
i 属于前景种子	$L(i)$	0
i 属于背景种子	0	$L(i)$
i 属于未确定区域	$S_{\text{fore}}(i)$	$S_{\text{back}}(i)$

2.3 改进算法的基本步骤

采用候选局部颜色模型, 并建立基于区域的交互式 Graph-Cut 分割框架, 算法的基本步骤如下.

1) 对输入图像进行 Mean-Shift 预分割, 建立 m 个候选局部颜色模型 $\{C_p\}_{p=1,2,\dots,m}$. 预分割的参数设置参考文献[7]所提供的数据. 如果存在过多误分割的效果(同一区域同时包含前景和背景), 可以通过调整参数减少误分割带来的错误.

2) 用户采用基于笔画的交互方式对输入图像进行前景和背景标记, 产生前景和背景种子集合 $\{F_s\}$ 和 $\{B_s\}$. 根据用户标记从候选局部颜色模型 $\{C_p\}_{p=1,2,\dots,m}$ 选取前景和背景颜色模型 θ_F 和 θ_B .

3) 将预分割后的区域作为图 G 的顶点, 建立基于区域的交互式 Graph-Cut 分割框架. 计算顶点 i 与前景和背景种子在颜色特征空间上的最短距离 d_i^F 和 d_i^B :

$$d_i^F = \min_{\text{num}(F)} \|M(i) - \theta_F\|, \\ d_i^B = \min_{\text{num}(B)} \|M(i) - \theta_B\|.$$

4) 根据表 1, 计算 3 种类型顶点区域项的值. 其中, $S_{\text{fore}}(i)$ 和 $S_{\text{back}}(i)$ 区域值的计算方法为:

$$S_{\text{fore}}(i) = \mu \cdot \text{neigh}(i) \cdot P(i | \theta_F), \\ S_{\text{back}}(i) = \mu \cdot \text{neigh}(i) \cdot P(i | \theta_B).$$

其中, 顶点属于前景和背景的概率 $P(i | \theta_F)$ 和 $P(i | \theta_B)$ 按照以下方法计算:

$$P(i | \theta_F) = \frac{d_i^B}{d_i^B + d_i^F + 1}, \\ P(i | \theta_B) = \frac{d_i^F}{d_i^B + d_i^F + 1}.$$

自适应因子 μ 根据式(2)、(3)计算得到, 常量因子 δ 取为 0.8. 同样, 根据式(4)计算边界项的能量.

5) 通过最大流原理, 求解能量函数(1)的最小值, 求得分割结果.

6) 用户根据分割的结果, 决定是否添加新的标记或者删除之前的标记. 如果分割结果达到用户的要求, 则算法结束, 输出分割结果. 如果用户仍然对分割结果不满意, 则添加新的标记, 或者删除之前的标记, 并转到 2).

通过基本流程可以看出, 本文算法建立在基于区域的交互式 Graph-Cut 分割框架上, 计算量比基于像素的 Graph-Cut 算法大大减少. 且在整个交互过程只需要建立一次图像的分布模型, 即候选局部颜色模型. 这些都有利于分割算法满足实时交互.

3 实验结果分析

为了验证改进算法的有效性, 在 2.20 GHz

CPU, 1 GB内存的 PC 机上,通过 Microsoft VC++ 6.0 实现本文算法. 利用 Berkeley 图像分割图库 (<http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench/>) 和文献[8]提供的测试图库进行测试,从交互延迟时间、分割错误率及自适应情况分析算法的性能.

3.1 交互延迟时间对比

假设一幅图像的像素点个数为 N , 像素相邻关系对应的总边数为 E , 则采用像素级的 Graph-Cut 求解所耗费的时间复杂度为 $O(NE^2)^{[9-10]}$. 图像预处理后的区域个数为 m , 边数为 e , Mean-Shift 算法预处理时间为 $O(N^2)$. 交互延迟时间表示用户标记之后 Graph-Cut 求解所耗费的时间, 如果每次交互操作都需要预处理, 则交互延迟时间的复杂度为 $O(N^2 + me^2)$. 一般情况下, 边数与顶点数的比例为常数, 则原算法的时间复杂度为 $O(E^3)$, 新算法为 $O(N^2 + e^3)$, 即 $O(e^3)$. 由于预处理后所计算的顶点数和边数明显减少, 与原算法的时间复杂度相比, 新算法的执行效率更高. 当图像分辨率提高时, 图像的像素个数增加, 但区域数不变, 新算法的效率提高得更显著. 另外, 本文算法一般只需要采用一次 Mean-Shift 分割就可完成预处理, 进一步减少了交互过程的计算量.

将本文算法的交互延迟时间与文献[2]及 Grabcut 算法进行对比, 3 个算法所对应的延迟时间如表 2 所示.

表 2 交互延迟时间对比

Table 2 Comparison of interactive lag time s			
图像	文献[2]	Grabcut	本文
143 090	1.079	3.703	0.016
doll	1.891	3.563	0.015
GT01	2.547	6.531	0.031
GT19	4.141	7.703	0.047
65019	1.219	2.125	0.015
85048	1.141	2.297	0.002
carsten	3.828	8.031	0.156
227092	1.203	2.141	0.001
stone2	2.219	3.922	0.032

文献[2]算法和本文算法的用户交互形式是一样的, 都是采用基于笔画的形式进行的. 而 Grabcut 算法则先采用矩形框将分割对象框住, 先自动迭代地完成初步分割. 这里 Grabcut 算法的交互延迟时间表示为初步分割的时间. 通过以上对比实验, 发现改进后的算法只需要很短的交互延迟时间, 达到了实时交互的效果.

3.2 错误率及自适应情况

为了验证改进后的算法也能达到分割效果精确, 在用户标记一样的情况下, 将本文算法与文献[2]算法的分割错误率进行比较分析. 对分割图库的图像进行分割, 分割错误率的对比结果如图 1 所示, 其具体的错误率见表 3. 最终统计出文献[2]算法的平均错误率为 1.95%, 而本文算法的平均错误率为 1.01%.

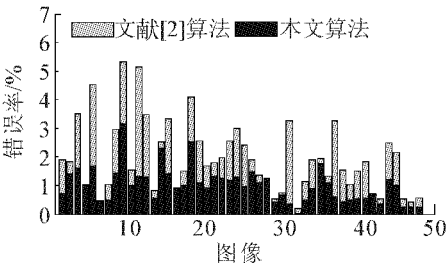


图 1 分割的错误率对比

Fig.1 Comparison of error rates

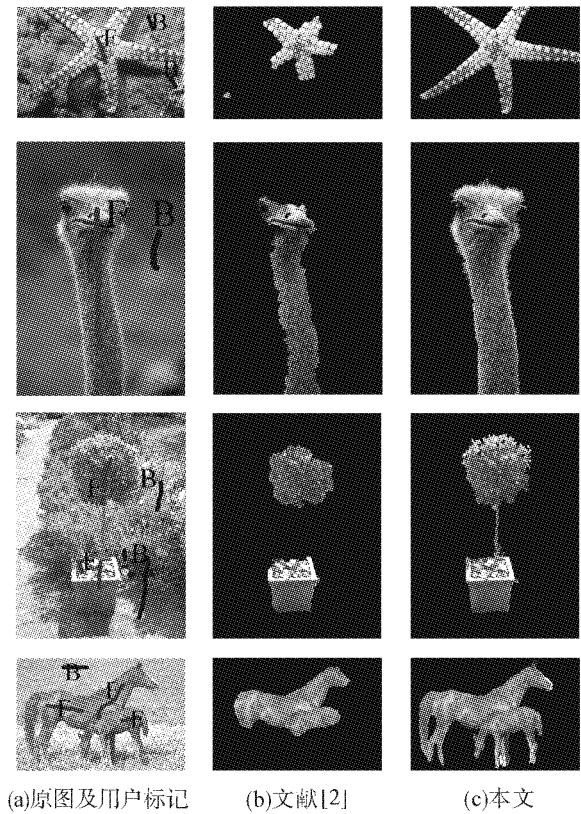


图 2 “shrinking bias”现象对比 (F 和 B 对应前景和背景标记)

Fig.2 Comparison of the “shrinking bias” phenomenon

Graph-Cut 算法会产生“shrinking bias”错误^[5], 不能准确分割含细长对象的图像, 影响分割的精确性. 对此, 比较了文献[2]算法与本文算法对含有细长对象图像的分割错误率. 图 2 列举了其中的 2 组对比实验结果, 可以看出本文算法可以在一定程度上缓解“shrinking bias”现象.

改进算法增加了自适应因子,可以适用于大部分图像的分割(其中本文 λ 值设置为 0.8). 图 3 列

举了各种不同形状和类型图像分割的结果. 实验表明,本文算法可以自适应于各种不同类型的图像.

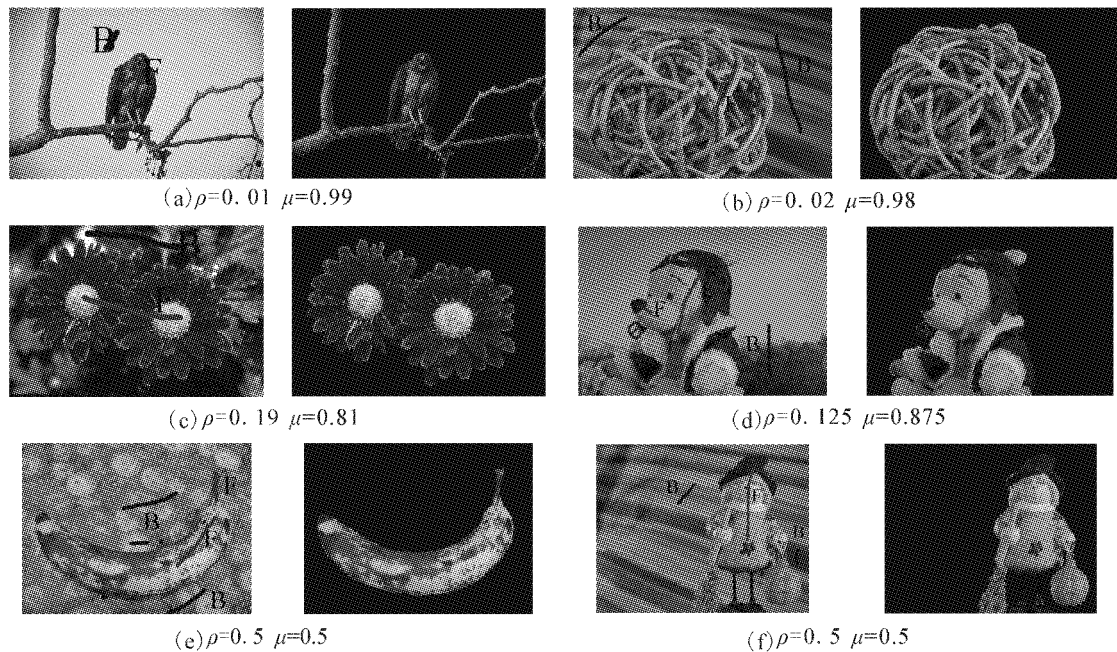


图 3 不同类型图像的分割结果

Fig 3 The segmentation results of different types of images

表 3 分割错误率对比

Table 3 Comparison of error rates

%

图像	文献[2]	本文	图像	文献[2]	本文	图像	文献[2]	本文	图像	文献[2]	本文
21077	1.91	0.71	208001	0.83	0.56	bool	2.41	0.96	person1	3.25	0.61
24077	1.83	1.39	209070	2.51	2.31	bush	1.91	1.46	person2	1.54	0.41
37073	3.49	1.60	227092	3.32	1.38	ceramic	1.36	1.11	person3	1.02	0.50
65019	0.47	1.04	271008	0.82	0.92	cross	1.24	1.24	person4	1.51	0.52
69020	4.51	1.68	304074	1.50	0.99	dol	0.54	0.43	person5	1.83	0.55
86016	0.37	0.44	326038	4.09	2.51	elefant	0.74	0.66	person6	0.71	0.71
106024	1.02	0.48	376043	2.56	1.12	flower	3.23	0.36	person7	0.52	0.34
124084	2.97	1.44	388016	1.67	0.93	fullmoon	0.20	0.01	person8	2.50	1.22
153077	6.03	3.14	banana1	1.78	1.33	grave	1.15	0.50	scissors	2.14	0.99
153093	1.54	1.01	banana2	1.97	1.26	llama	1.90	0.88	sheep	0.54	0.23
181079	5.12	1.34	banana3	2.55	1.19	memorial	1.95	1.75	stone1	0.40	0.23
189080	3.48	1.28	book	3.01	1.30	music	1.33	1.10	stone2	0.56	0.24

4 结束语

本文采用局部颜色模型对前景和背景颜色分布进行估计,并建立了基于区域的交互式 Graph-Cut 分割的框架. 实验表明这一改进是有效的,不仅可以更好地估计颜色分布,保持了分割效果的精确性,而且每次迭代不需要重新建立分布,减少了交互延迟时间,达到了实时交互的效果. 但是当前景与背景存在过多相似颜色时,与其他算法一样,本文算法也会产生比较大的分割错误率,今后将针对这一问题做

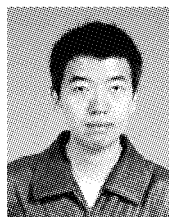
进一步的改进.

参考文献:

- [1] KASS M, WITKIN A, TERZOLPOULOS D. Snakes: active contour models[J]. International Journal of Computer Vision, 1988, 1(4): 321-331.
- [2] YUNRI Y, JOLLY B M P. Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images [C]//Proceedings of International Conference on Computer Vision. Vancouver, Canada, 2001, 1: 105-112.
- [3] ROTHER C, KOLMOGOROV V, BLAKE A. Grabcut-in-

- teractive foreground extraction using iterated graph cuts [J]. ACM Transactions on Graphics, 2004, 23(3): 309-314.
- [4] LI Yin, SUN Jian, TANG Chikeung, et al. Lazy snapping [C]//Computer Graphics Proceedings, Annual Conference Series (ACM SIGGRAPH). Los Angeles, USA, 2004: 303-308.
- [5] VICENTE A, KOLMOGOROV V, ROTHER C. Graph cut based image segmentation with connectivity priors [C]//Proceeding of IEEE International Conference on CVPR. [S. l.], 2008: 1-8.
- [6] 刘陈, 李凤霞, 张艳. 图割与泛形信息结合[J]. 计算机辅助设计与图形学学报, 2009, 21(12): 1753-1760.
LIU Chen, LI Fengxia, ZHANG Yan. An interactive object cutout algorithm based on graph-cut and generalized shape prior[J]. Journal of Computer-aided Design & Computer Graphics, 2009, 21(12): 1753-1760.
- [7] COMANICIU D, MEER P. Mean shift: a robust approach toward feature space analysis [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002, 24(5): 603-619.
- [8] RHEMANN C, ROTHER C, WANG J. A perceptually motivated online benchmark for image matting [C]//Proceedings of IEEE International Conference on CVPR. 2009: 1826-1833.
- [9] YUNRI Y, BOYKOV, KOLMOGOROV V. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2004, 26(9): 1124-1137.
- [10] KOLMOGOROV V, ZABIH R. What energy functions can be minimized via graph cuts [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2004, 26(2): 147-159.

作者简介:



郑加明,男,1986年生,硕士研究生,主要研究方向为图形图像处理。



陈昭炯,女,1964年生,教授,主要研究方向为图形图像处理与智能算法设计等,主持及参与多项国家和省级基金项目,发表学术论文50余篇。

2011 第6届智能系统与知识工程国际会议 (ISKE 2011) 2011 International Conference on Intelligent Systems and Knowledge Engineering (ISKE 2011)

2011 第6届智能系统与知识工程国际会议 (ISKE 2011) 将于2011年12月15日~17日在上海召开。已成功在06年上海、07年成都、08年厦门、09年成都、10年杭州召开。此次会议将由上海交通大学主办,加州州立大学,西南交通大学,比利时核研究中心等协办。本次会议将征集有关智能系统、知识工程等领域的论文,主要范围包括(但不限于):智能计算,机器学习,模式识别,知识发现,模糊推理与概率推理,自然语言处理,语音及图像处理及识别,知识管理,内容管理,多agent系统,新型智能计算(如分布式计算,云计算、服务计算、普适计算等),智能控制,智能系统与知识工程在工业、商务、医疗、科学及政务等领域的应用。所有被录用的论文将被 Ei Compendex 与 ISTP 全文检索。

主要范围包括(但不限于):智能计算,机器学习,模式识别,知识发现,模糊推理与概率推理,自然语言处理,语音及图像处理及识别,知识管理,内容管理,多agent系统,新型智能计算(如分布式计算、云计算、服务计算、普适计算等),智能控制,智能系统与知识工程在工业、商务、医疗、科学及政务等领域的应用。所有被录用的论文将被 Ei Compendex 与 ISTP 全文检索。部分优秀论文推荐到 EI 期刊发表。现在您提交论文后20天左右,我们将给您发出是否录用的通知。

Web site: <http://iske2011.sjtu.edu.cn>.