

抓取剪切 GrabCut——交互式迭代前景分离算法

数字信号处理期末作业

傅长青 卢川

2016 年 6 月 13 日

摘 要

本论文实现了 Rother et al. 2004 年的前景分割算法 [2], 大体是总结了以往的若干算法, 特别是贝叶斯聚类算法 Bayes Matting (Chuang et al. 2001, Ruzon and Tomasi 2000), 和图像切割算法 Graph Cut (Boykov and Jolly 2001; Greig et al. 1989), 并做出了一些创新, 包括“迭代优化”和“不完全标记”. 此算法主要应用统计模型中的 EM 算法, 以及一些统计物理的思想.

1 引言

1.1 以往研究

Magic wand; Intelligent scissors; Bayes matting (提出了 Trimap 模型); Knockout 2; Graph Cut (和 Bayes matting 类似, 包括 Trimap 和概率颜色模型. 将在 section 2 详细说明. 这种方法可以处理前景背景渐进色); Level sets 等.

1.2 抓取剪切 Grabcut:

1.2.1 符号定义

$T = \{T_B, T_F, T_U\}$ Trimap; T_B : Background, T_F : Foreground, T_U : Undecided

$z = (z_1, \dots, z_N)$ 图像灰度值

$\underline{\alpha} = (\alpha_1, \dots, \alpha_N)$: 前景的可能性. $\alpha \in \{0, 1\}$ 为硬分割, $\alpha \in [0, 1]$ 为一般情况

$\underline{\theta} = \{h(z; \alpha); \alpha = 0, 1, \int_z h(z; \alpha) = 1\}$ 图像前景背景的灰度值分布. 由灰度值的频率 (histogram) 组成.

$U(\underline{\alpha}, \underline{\theta}, z) := \sum_n -\log h(z_n, \alpha_n)$ 已知灰度分布频率 $\underline{\theta}$ 时, $\underline{\alpha}$ 对数据 z 的拟合程度.

$V(\underline{\alpha}, z) := \gamma \sum_{(m,n) \in C} \|m - n\|^{-1} [\alpha_m \neq \alpha_n] \exp(-\beta(z_m - z_n)^2)$ 描述图像的光滑程度. 其中 C 是相邻像素对 (回忆扫雷游戏), $[\alpha_m \neq \alpha_n] := 1_{\alpha_m \neq \alpha_n}(m, n)$, γ 是常数

1.2.2 思路

理想情况下, 使 α 值在 T_U 上连续, 不加以 α 只能取 0 或 1 这样的限制. 这样一来烟雾、头发之类的物体可以自动处理. 然而这样的一般方法 (Ruzon 2000, Chuang 2001) 会造成保护色或靠近颜色下的误判情况. 故我们使用下列步骤一步一步抓取前景.

首先考虑硬分割 ($\alpha \in \{0, 1\}$), 使用迭代 Graph Cut 方法 (Section 2, 3), 其次用边界聚类计算一个窄带 (Section 4). Grabcut 并不处理边缘之外的完全透明区域, 如果要处理可以用 Matting Brush (Chuang 2001) 方法, 但据经验这种方法还是只能处理边界分明的情况.

Grabcut 的创新在于用了两个方法: 迭代估计和不完全标记. 以及使用了一个计算 α 值的新方法用于边界聚类.

2 基于 Graph Cut 的图像分割

分割的目标是从 z 和 θ 推断未知的 α . 定义 Gibbs 能量:

$$\mathbf{E}(\underline{\alpha}, \underline{\theta}, \mathbf{z}) = U(\underline{\alpha}, \underline{\theta}, \mathbf{z}) + V(\underline{\alpha}, \mathbf{z})$$

其中, $U(\underline{\alpha}, \underline{\theta}, z) := \sum_n -\log h(z_n, \alpha_n)$ 是已知灰度分布频率 $\underline{\theta}$ 时, $\underline{\alpha}$ 对数据 z 的拟合程度. $\beta = 0$ (处处视为光滑) 时为所谓的 Ising 先验. 这里我们取 $\beta = (2\mathbb{E}(z_m - z_n)^2)^{-1}$ (Boykov and Jolly 2001). 然后取全局最小为划分的估计:

$$\hat{\underline{\alpha}} = \arg \min_{\underline{\alpha}} \mathbf{E}(\underline{\alpha}, \underline{\theta})$$

首先, 用混合高斯分布模型 (GMM) 代替灰度出现频率; 第二, 将单次最小切割算法用迭代算法代替. 第三, 需要用户交互解决的地方用不完全标记法即可放宽要求, 即用套索或一个长方形将物体框住即可.

3 GrabCut 分割算法

分为迭代估计 (译注: 一般称为 EM 算法) 和不完全标记两个部分. 下面开始像素空间为 RGB 彩色, 采取软分割的方法 (Ruzon2000; Chuang 2001)

定义 $\mathbf{k} = \{k_1, \dots, k_N\}$ 现在 Gibbs 能量变成:

$$\mathbf{E}(\underline{\alpha}, \mathbf{k}, \underline{\theta}, \mathbf{z}) = U(\underline{\alpha}, \mathbf{k}, \underline{\theta}, \mathbf{z}) + V(\underline{\alpha}, \mathbf{z})$$

U 是 K 元混合高斯模型 (GMM), K 一般取 5 即可.

$$U(\underline{\alpha}, \underline{\theta}, \mathbf{z}) := \sum_n^K D(\alpha_n, k_n, \underline{\theta}, z_n)$$

其中

$$\begin{aligned}
D(\alpha_n, k_n, \underline{\theta}, z_n) &= -\log p(z_n | \alpha_n, k_n, \underline{\theta}) - \log \pi(\alpha_n, k_n) \\
&= -\log \pi(\alpha_n, k_n) + \frac{1}{2} \log \det \Sigma(\alpha_n, k_n) + \frac{1}{2} [z_n - \mu(\alpha_n, k_n)]^T \Sigma(\alpha_n, k_n)^{-1} [z_n - \mu(\alpha_n, k_n)]
\end{aligned}$$

p 是高斯分布密度函数, π 是混合分布权值系数. 现在系数 $\underline{\theta}$ 为:

$$\underline{\theta} = \{\pi(\alpha, k), \mu(\alpha, k), \Sigma(\alpha, k), \alpha = 0, 1, k = 1, \dots, K\}$$

光滑项:

$$V(\underline{\alpha}, z) := \gamma \sum_{(m, n \in C)} [\alpha_m \neq \alpha_n] \exp(-\beta \|z_m - z_n\|^2)$$

算法 1 GrabCut

初始化:

用户给定 T_B , 设定 $T_F = \emptyset$, $T_U = \overline{T_B}$, $n \in T_B$ 处 $\alpha_n = 0$, $n \in T_U$ 处 $\alpha_n = 1$

迭代步骤:

1. $k_n := \arg \min k_n D_n(\alpha_n, n, \underline{\theta}, z_n)$
2. $\underline{\theta} = \arg \min_{\underline{\theta}} U(\underline{\alpha}, \mathbf{k}, \underline{\theta}, z)$
3. $\{\alpha_n : n \in T_U\} = \arg \min_{\mathbf{k}} \mathbf{E}(\underline{\alpha}, \mathbf{k}, \underline{\theta}, z)$
4. 重复 Step 1.

用户交互

编辑: 固定一些像素的 α 值为 0 (称作背景刷) 或 1 (前景刷), 运行一次 Step 3.

提炼: (可选步骤) 进行整个迭代步骤.

注: 论文后面还讨论了透明像素点的处理, 不属于 GrabCut 算法范畴, 故略去不讨论.

4 实验结果

采用 OpenCV [1] x Python, 实验结果如下, 示例采用了著名的 Lena (credit: shot by photographer Dwight Hooker, cropped from the centerfold of the November 1972 issue of Playboy magazine.):

5 总结与感想

目前图像处理的研究中 (参考 <http://ipol.im>) 除了分解算法外, 也有不少基于统计学习方法这也是这篇报告采用的方法. 今后的学习中我们将会尝试信号处理等方法解决相关问题.

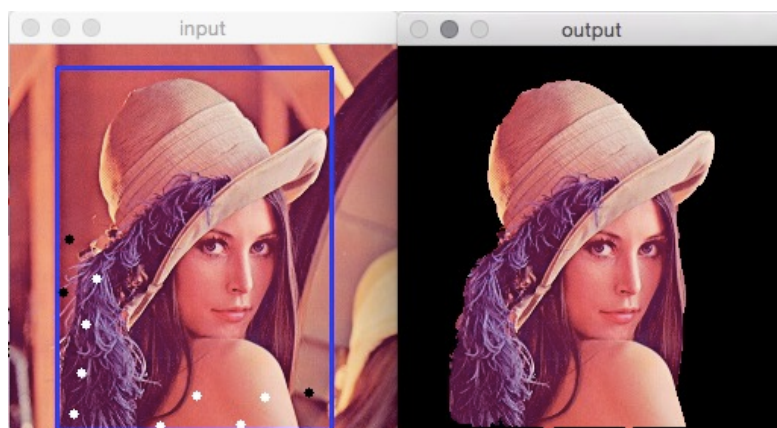


图 1: 实验结果

参考文献

- [1] Gary Bradski and Adrian Kaehler. *Learning OpenCV: Computer vision with the OpenCV library*. " O'Reilly Media, Inc.", 2008.
- [2] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. In *ACM transactions on graphics (TOG)*, volume 23, pages 309–314. ACM, 2004.