Introduction and Realization to GrabCut, A Foreground-Background Seperation Algorithm Using Iteration and Interaction

Term Project for DSP

Chuan Lu, Changqing Fu

2016-06-21

Abstract

In this project, we realized the foreground-background seperation algorithm proposed by Rother et al. [2004]. It mainly concluded some seperation algorithms in the past years, especially the Bayes clustering algorithm Bayes Matting (Chuang et al. [2001], Ruzon and Tomasi [2000]) and image cutting algorithm GraphCut (Boykov and Jolly [2001]; Greig et al. [1989]), and made some improvement, including 'Iterative Improvement' and 'Incomplete Tag'. This algorithm mainly used the Expectation Maximization algorithm in statistical models, and some ideas from statistical physics.

1 Introduction

1.1 Some Past Algorithms

Magic wand; Intelligent scissors; Bayes matting(Proposed Trimap model); Knockout 2; Graph Cut(Similar with Bayes matting, including Trimap and probability color model, will be detailedly expressed in section 2. This model can handle the slowly changing color between foreground and background); Level sets, etc.

1.2 Grabcut:

1.2.1 Definition

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

$z = (z_1, \ldots, z_N)$: Grey scale of an image

$\underline{\alpha} = (\alpha_1, \ldots, \alpha_N)$: The possibility of a point being in the foreground region. $\alpha \in \{0, 1\}$ is the hard-cut case, $\alpha \in [0, 1]$ is the general case.

$\underline{\theta} = \{h(z;\alpha); \alpha = 0, 1, \int_z h(z;\alpha) = 1\}$: The distribution of grey scale in background and foreground, it is determined by the histogram of grey scale.

$U(\underline{\alpha}, \underline{\theta}, z) := \sum_n -\log h(z_n, \alpha_n)$: When knowing the distribution of grey scale$\underline{\theta}$, the fitting degree of $\underline{\alpha}$ to the data z.

$V(\underline{\alpha}, z) := \gamma \sum_{(m,n\in C)} \|m - n\|^{-1} [\alpha_m \neq \alpha_n] \exp(-\beta(z_m - z_n)^2)$: The degree of smoothness inside the image, in which C is the neighbouring pixel pairs (like the Minesweeper), $[\alpha_m \neq \alpha_n] := 1_{\alpha_m \neq \alpha_n}(m, n)$ γ is a constant.

1.2.2 Ideas

Ideally, we let α be constant in T_U without constraint such as α need to be chosen from $0, 1$. In this case tiny objects like smoke and hairs can be handled automatically. However these methods (Ruzon [2000] Chuang [2001]) will lead to misjudgement in gradually changing colors. Thus we use the following steps to "Grab" the foreground step by step.

First, consider the hard-seperation ($\alpha \in 0, 1$), use the Iterative Graph Cut method (Section 2, 3). Then compute a narrow band using border clustering. GrabCut does not handle the completely transparent zone outside the border; if need so, you can use Matting Brush Algorithm (Chuang [2001]), but according to experience it can only handle the case with a clear border.

The innovation of GrabCut is in its using two three methods: Iterative Estimation, Incomplete Tagging, and a new method for computing α, used in border clustering.

2 Image Cutting based on GraphCut

Our Goal for seperation is to infer the unknown $\underline{\alpha}$ using z and $\underline{\theta}$. Define the Gibbs Energy:

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

in which $U(\underline{\alpha}, \underline{\theta}, z) := \sum_n -\log h(z_n, \alpha_n)$ is the fitting degree of $\underline{\alpha}$ to the data z when knowing the distribution of grey scale $\underline{\theta}$. $\beta = 0$ (smooth everywhere) is the so-called Ising Prior. In our application we choose $\beta = (2\mathbb{E}(z_m - z_n)^2)^{-1}$ (Boykov and Jolly [2001]). The we choose the estimation s.t. reaching the global minimum:

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

Firstly, we use Gaussian Mixed Model (GMM) in place of grey scale probability; Secondly, use an iterative algorithm in place of one minimizing-cut algorithm. Thirdly, use incomplete tagging to solve the points needing user interaction, which means to use a rectangle to frame the foreground object.

3 GrabCut seperation algorithm

It is devided into Iterative Estimation (the EM algorithm) and incomplete tagging. The data space is RGB 3-dimension Euclidean space, and we use soft-cutting methods (Ruzon [2000]; Chuang [2001]).

Define $\mathbf{k} = \{k_1, \ldots, k_N\}$. Then the Gibbs Energy is:

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

U is a GMM with k components. In most cases $k = 5$.

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

in which

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

p is the density function of Gaussian distribution, $\pi$is the weight of mixed distribution. Now the coefficient $\underline{\theta}$ is

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

And the smoothing term

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

Algorithm 1 GrabCut

Initialize:

With user interaction given T_B, let $T_F = \emptyset \; T_U = \overline{T_B}$, $\alpha_n = 0$ where $n \in T_B$, $\alpha_n = 1$ where $n \in T_U$.

Iteration:

1. $k_n := \arg\min k_n D_n(\alpha_n, k_n, \theta, z_n)$
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. Repeat Step 1.

User Interaction:

Edit: Fix the α value of some pixels to 0 (background) or 1 (foreground), then execute Step 3.

Optimize: Execute the whole iteration again.

---

4 Realization and Results

We used OpenCV and Python 3.0 to implement this algorithm. In average, Python needs 30 seconds to execute the broder-clustering part, where we used Max Flow/Min Cut algorithm to do so. As an example, we used the famous Lena (credit: Dwight Hooker, Nov 1972 Playboy.)

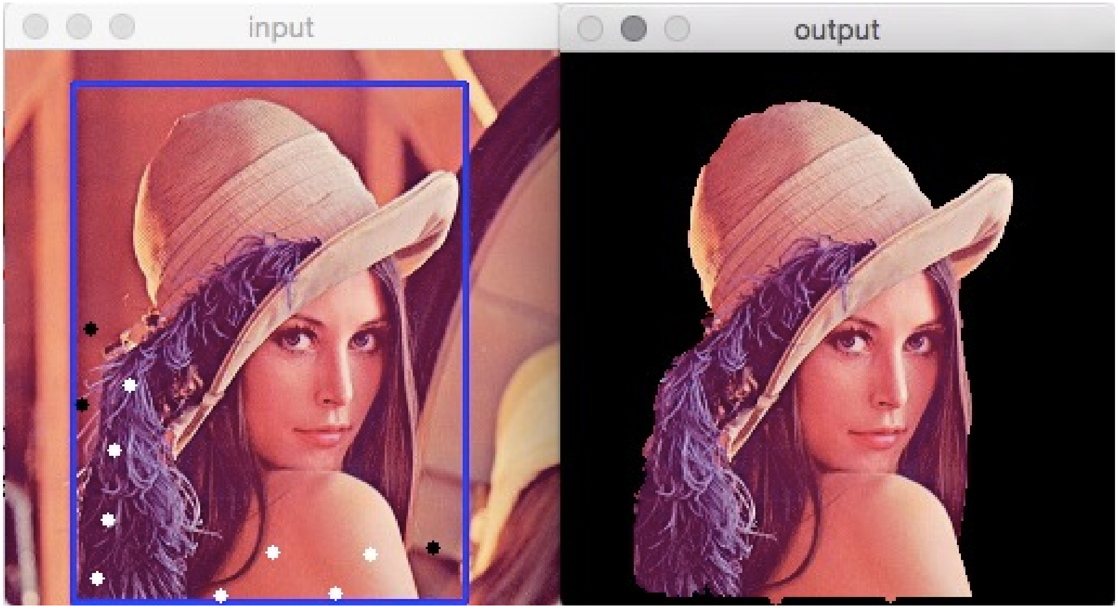

Figure 1: Result

5 Conclusion

In state-of-the-art digital image processing research (reference: http://ipol.im), there are many algorithms based on statistical learning models. Later we should try some signal processing methods in digital image processing.