Building Detection Using Graph Cut

12 mai 2014

Author :
LI Jia
XU Minmin

Course :
Méthodes avancées de traitement d'images

1. Grab cut
   1. approach
   2. synthetic test
2. Automation
   1. shadow detection
   2. vegetation detection
   3. ROI region
   4. Foreground region

Motivation

- Application : urban monitoring, change detection, estimation of human population, etc.
- Challenging : complex background, without human interacting.

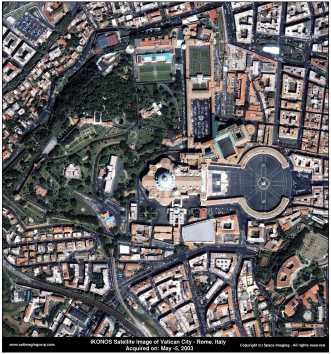

FIGURE 1: Example of Arian Urban Image

Intuition

Approach proposed by Ok *et. al* [**?**] :

- Grab cut (Rother [**?**]) : semi-automatic segmentation, with a user defined foreground-background window.

FIGURE 2: Grab cut method illustration

- Foreground-background estimation : prior knowledge, shadow, vegetation ...

1. **Grab Cut**
   1. Approach
   2. Experience
2. Automation
   1. shadow detection
   2. vegetation detection
   3. ROI region
   4. Foreground region

Image Segmentation by Graph Cut

Gibb's energy for image segmentation :

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

- $\mathbf{z} = (z_1, z_2, ..., z_N)$, a N-pixel image.
- $\alpha = (\alpha_1, \alpha_2, ..., \alpha_N)$ label for each pixel, typically $\alpha_n \in \{0, 1\}$.
- θ, background model for each label/class, empirical histogram, GMM...

Image Segmentation by Graph Cut

Gibb's energy for image segmentation :

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

- $U = \sum_n -\log(p_\theta(\alpha_n, z_n))$ the likelihood term.
- $V = \gamma \sum_{n,m \in \mathbf{C}} [\alpha_m \neq \alpha_n] \exp(-\beta |z_n - z_m|^2)$ the regularity term, $\beta = 0$ correspond to the Isings model.

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

- Standard graph cut for minimization. ([Boykov and Jolly 2001 ; Kolmogorov and Zabih 2002])

Grab Cut

Grab cut for foreground background separation :

- GMM for the likelihood term. One GMM for the foreground and the other for the background.
- Iterative estimation and parameters learning instead of a one-shot minimization.
- Relaxed user interactive labeling.

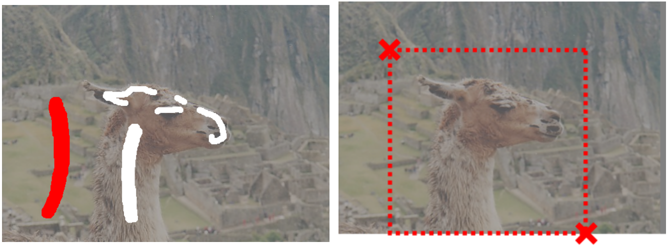

FIGURE 3: Illustration of Graph cut and Grab cut labeling. (Rother 2004 [?])

Grab Cut

Energy for grab cut :

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

- $\mathbf{k} = (k_1, k_2, ..., k_N)$, with $k_n \in \{1, ..., K\}$ assigning each pixel to a GMM component.
- $U(\alpha, \mathbf{k}, \theta, \mathbf{z}) = \sum_n D_n(\alpha, \mathbf{k}, \theta, \mathbf{z})$, with

$$\begin{aligned} D_n(\alpha, \mathbf{k}, \theta, \mathbf{z}) &= -\log(p_\theta(\alpha_n, k_n, z_n)) - \log(\pi(\alpha_n, k_n)) \\ &= -\log(\pi(\alpha_n, k_n)) - 1/2 \log \det \Sigma(\alpha_n, k_n) \\ &- 1/2[z_n - \mu(\alpha_n, k_n)]^T \Sigma(\alpha_n, k_n)^{-1} [z_n - \mu(\alpha_n, k_n)] \end{aligned}$$

Minimization Algorithm

1. *Initialization*. $\alpha_n = 1$ for $n \in \mathbf{T}_u$, $\alpha_n = 0$ otherwise.
2. *Learning*. $\mathbf{k} = \arg\min_{\mathbf{k}} U(\alpha, \mathbf{k}, \theta, \mathbf{z})$, $\theta = \arg\min_{\theta} U(\alpha, \mathbf{k}, \theta, \mathbf{z})$
3. *Estimation*. $\min_{\alpha} \min_{\mathbf{k}} \mathbf{E}(\alpha, \mathbf{k}, \theta, \mathbf{z})$
4. *Iteration*. Repeat 2,3 until convergence.

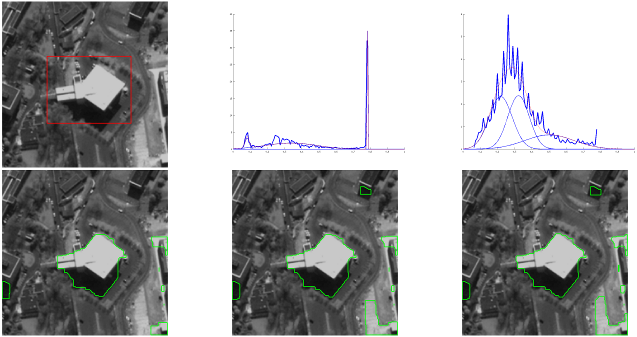

FIGURE 4: Illustration of the algorithm

1. Grab Cut
   1. Approach
   2. Experience
2. **Automation**
   1. shadow detection
   2. vegetation detection
   3. Foreground region
   4. ROI region

shadow detection

Two steps : K-means and growth of region

- Use K-means to find the first peak and therefore find a simple threshold for shadows. Normally 5 or 6 cluster would be enough for the detection.
- Use the result of threshold as seed to apply method of growth of region to obtain the complete shadows. We need to control the tolerance in this algorithm.

FIGURE 5: Shadow detection

vegetation detection

- Likelihood between the color and the color of vegetation.

Clean shadows

- First remove the detected vegetation from the shadow.
- Remove the shadow corresponding to the detected vegetation by double threshold of fuzzy map on each connected component.

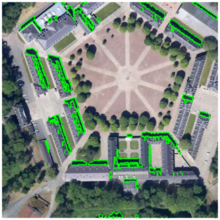

FIGURE 6: Shadow before and after being cleaned

Foreground detection

- Construct shadow structuring element with reversed sun direction and length of shadow. $\nu_{L,\lambda}$, L length and λ direction.
- Construct non-flat structuring element $\nu_{L,\lambda,\sigma,\kappa} = \nu_{L,\lambda} * \nu_{\sigma,\kappa}$, with κ kernel size and σ rate of decrease.
- Create the fuzzy map $\beta_\lambda(B) = (B^{per} \oplus \nu_{L,\lambda,\sigma,\kappa}) \cap B^C$and then apply the double threshold θ_1, θ_2 to obtain the foreground mask.

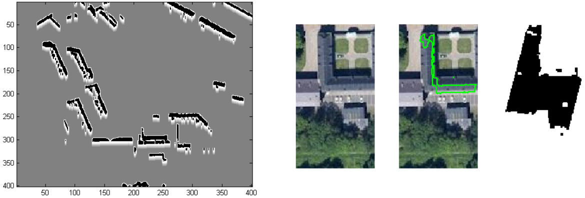

FIGURE 7: Fuzzy map and Foreground detection

ROI detection

- Dilate the shadow with the shadow structural element in order to obtain the regions of interest(in which there are buildings we want to detect). We need to control the length L of the element in order not to exclude part of building.

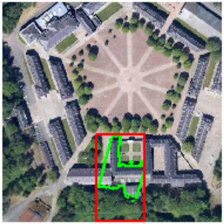

FIGURE 8: Region of interest

Thank you !

Q&A

References