Combinatorial Optimization in Computer Vision (IN2245)

Frank R. Schmidt
Csaba Domokos

Winter Semester 2015/2016

24. Object Segmentation with Prior Information

GrabCut revisited

We have already discussed the **GrabCut** method (cf. Lecture 6). Let $I : \Omega \to \mathbb{R}^d$ be an image into a d-dimensional color space.

1. The user provides a *bounding box* around the object.
2. With respect to this *bounding box*, probability models p for foreground and q for background are estimated (using Gaussian mixture models).
3. The data term of a pixel i is set to $f_i := \log\left(\frac{q(I_i)}{p(I_i)}\right)$. The length term between two pixels is set to $c(i,j) := \lambda \exp\left(-\frac{|I(i)-I(j)|^2}{2\sigma^2}\right)$.
4. Minimize the energy for $x \in \mathbb{B}^N$

$$E(x) = \sum_{i=1}^{N} f_i x_i + \sum_{i=1}^{N} \sum_{j \in \mathcal{N}(i)} c(i,j) x_i \bar{x}_j$$

   and obtain a cut (S,T).
5. Update p and q with respect to S and T. If p or q changes go to `Step 3`.
6. S provides for an image segmentation.

GrabCut in one cut

Object segmentation

The basic *object segmentation energy* combines **boundary regularization** with **log-likelihood ratios** for fixed foreground and background appearance models, e.g. color distributions, θ^1 and θ^0:

$$\begin{aligned} E(S \mid \theta^1, \theta^0) &= -\sum_{i \in \Omega} \log P(I(i) \mid \theta^{s_i}) + |\partial S| \\ &= -\sum_{i \in \Omega} \log P(I(i) \mid \theta^{s_i}) + \sum_{(i,j) \in \mathcal{N}} w_{ij} |s_i - s_j| \end{aligned}$$

where Ω is the set of all pixels. $s_i \in \mathbb{B}$ are indicator variables for segment $S \subset \Omega$ and $\mathcal{N}$ is the set of all pairs of neighboring pixels.

There are efficient methods for global minimization of E, e.g., **graph cut**.

In many applications, however, the appearance models may not be known *a priori*:

$$E(S, \theta^1, \theta^0) = -\sum_{i \in \Omega} \log P(I(i) \mid \theta^{s_i}) + |\partial S| \,.$$

Histogram *

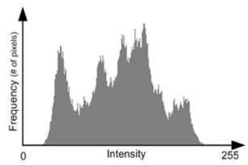
Let us assume that we are given a set of samples $x_1, x_2, \ldots, x_n$ drawn from a probability distribution of a continuous random variable X.

Assume that a *partition* of the range of X is fixed, that is the entire range of values is divided into a series of (equal length) intervals, called **bins**.

We count how many values fall into each interval. The **frequency** k_i is the number of samples corresponding to the i^{th} bin.

If the *bins* are of equal size h, then a rectangle is erected over the *bin* for each *bin* $i = 1, \ldots, K$ with height

$$\tilde{p}_i = \frac{k_i}{h} \,.$$



http://www.songho.ca/dsp/histogram/files/histogram01.png

This graph is called **(unnormalized) histogram** of the samples $x_1, x_2, \ldots, x_n$.

Normalized histogram *

The **normalized histogram** has *relative frequencies*

$$\frac{\tilde{p}_i}{n} = \frac{k_i}{nh} =: p_i \qquad \text{for all } i = 1, \ldots, K \,.$$

It shows the proportion of samples that fall into each of several categories, such that the sum of the area of rectangles equals to one:

$$\sum_{i=1}^{K} h \cdot \frac{k_i}{nh} = \frac{1}{n} \sum_{i=1}^{K} k_i = 1 \,.$$

It is worth noting that *bins*, in general, need not be of equal width. In that case, the erected rectangle has area proportional to the frequency of samples in the *bin*.

Equivalent formulation of $E(S, \theta^1, \theta^0)$

$$E(S, \theta^1, \theta^0) = -\sum_{i \in \Omega} \log P(I(i) \mid \theta^{s_i}) + |\partial S|$$

is known to be NP-hard.

Let θ^1 and θ^0 be represented by (non-parametric) **color histograms** θ^S and $\theta^{\bar{S}}$ for inside object S and background $\bar{S} = \Omega \backslash S$, respectively. Thus

$$\sum_{i \in \Omega} \log P(I(i) \mid \theta^{s_i}) = \sum_{i \in S} \log P(I(i) \mid \theta^S) + \sum_{i \in \bar{S}} \log P(I(i) \mid \theta^{\bar{S}}) \,.$$

The first term can be written as

$$\begin{aligned} \sum_{i \in S} \log P(I(i) \mid \theta^S) &= \sum_k \sum_{i \in S, I(i)=k} \log P(k \mid \theta^S) = \sum_k \sum_{i \in S, I(i)=k} \log p_k \\ &= \sum_k \tilde{p}_k \log p_k = |S| \sum_k \frac{\tilde{p}_k}{|S|} \log p_k = |S| \sum_k p_k \log p_k \\ &= -|S| \cdot H(\theta^S) \,. \end{aligned}$$

Equivalent formulation of $E(S, \theta^1, \theta^0)$

Putting together, we get that

$$\begin{aligned} E(S,\theta^1,\theta^0) &= -\sum_{i\in\Omega} \log P(I(i) \mid \theta^{s_i}) + |\partial S| \\ &= -\sum_{i\in S} \log P(I(i) \mid \theta^{S}) - \sum_{i\in \bar S} \log P(I(i) \mid \theta^{\bar S}) + |\partial S| \\ &= |S|\cdot H(\theta^S) + |\bar S| \cdot H(\theta^{\bar S}) + |\partial S| \; . \end{aligned}$$

This formulation is useful for analyzing the properties of the energy. The entropy terms here prefer segments with more peaked color distributions, which also imply distributions with small overlap.

Note that the global minimum of segmentation energy does not depend on the initial color models provided by the user.

Reminder: $H(p) = \sum_k p_k \log p_k$ stands for the **entropy** of a discrete random variable with probability distribution p.

Equivalent formulation of the entropy terms

We use Ω_k to denote the set of all pixels in color bin k, and $S_k = S \cap \Omega_k$ is a subset of pixels of color k inside object segment. For further analysis, let us rewrite equivalently the two entropy terms as

$$\begin{aligned} |S|\cdot H(\theta^S) + |\bar S| \cdot H(\theta^{\bar S}) &= -|S|\sum_k p_k \log p_k - |\bar S| \sum_k \bar p_k \log \bar p_k \\ &= -|S|\sum_k \frac{|S_k|}{|S|}\log\frac{|S_k|}{|S|} - |\bar S|\sum_k \frac{|\bar S_k|}{|\bar S|}\log\frac{|\bar S_k|}{|\bar S|} \\ &= \sum_k |S_k|(\log|S| - \log|S_k|) + \sum_k |\bar S_k|(\log|\bar S| - \log|\bar S_k|) \\ &= \log|S|\sum_k |S_k| + \log|\bar S|\sum_k|\bar S_k| - \sum_k |S_k|\log|S_k| - \sum_k |\bar S_k|\log|\bar S_k| \\ &= |S|\log|S| + |\bar S|\log|\bar S| - \sum_k |S_k|\log|S_k| - \sum_k |\bar S_k|\log|\bar S_k| \\ &= |S|\log|S| + |\Omega\backslash S|\log|\Omega\backslash S| - \sum_k |S_k|\log|S_k| - \sum_k |\Omega_k\backslash S_k|\log|\Omega_k\backslash S_k| \; . \end{aligned}$$

Color separation bias

We need to introduce the following notation for subset $B \subset A$:

$$h_A(B) = |B|\log|B| + |A\backslash B|\log|A\backslash B| \; ,$$

which is also known as the **Jensen-Shannon (JS) divergence**.

$$\begin{aligned} |S|\cdot H(\theta^S) + |\bar S| \cdot H(\theta^{\bar S}) =& |S|\log|S| + |\Omega\backslash S|\log|\Omega\backslash S| \\ & - \sum_k |S_k|\log|S_k| - \sum_k |\Omega_k\backslash S_k|\log|\Omega_k\backslash S_k| \\ =& h_\Omega(S) - \sum_k h_{\Omega_k}(S_k) \; . \end{aligned}$$

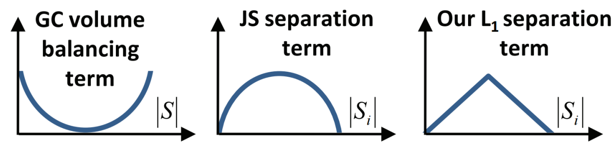
Reminder: If $H : \mathbb{R} \to \mathbb{R}$ is a concave function, then $E_H : 2^\Omega \to \mathbb{R}$ is *submodular* with $E_H(A) := H(|A|)$ (cf. Lecture 2).

One can see that $-h_{\Omega_k}(S_k)$ is *submodular* for all $k = 1,\dots,K$, nevertheless $h_\Omega(S)$ is *supermodular*, which makes optimization difficult.

L_1 separation term

$$|S|\cdot H(\theta^S) + |\bar S| \cdot H(\theta^{\bar S}) = h_\Omega(S) - \sum_k h_{\Omega_k}(S_k) \; .$$

The term $h_\Omega(S)$ is referred to as *volume balancing* term, and $-h_{\Omega_k}(S_k)$ is referred to as *JS color separation* term.



$h_\Omega(S)$ $-h_{\Omega_k}(S_k)$ $-|S_k - \bar S_k|$

Instead of $-h_{\Omega_k}(S_k)$, a more basic L_1 *separation* term is proposed to **replace** the *JS color separation* term. It results in a simpler graph construction with much fewer auxiliary nodes leading to higher efficiency.

Appearance overlap

Let θ^S and $\theta^{\bar S}$ the **unnormalized** color histograms for the foreground and background appearance, respectively. Let n_k^S and $n_k^{\bar S}$ be the number of the foreground and background pixels, respectively, in bin k.

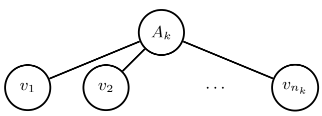
The *appearance overlap term* penalizes the intersection between the foreground and background bin counts:

$$\begin{aligned} E_{L_1}(\theta^S,\theta^{\bar S}) &= -\|\theta^S - \theta^{\bar S}\|_1 = -\sum_{k=1}^K |n_k^S - n_k^{\bar S}| \\ &= -\left(\sum_{k=1}^K \max(n_k^S, n_k^{\bar S}) - \min(n_k^S, n_k^{\bar S})\right) \\ &= \sum_{k=1}^K \min(n_k^S, n_k^{\bar S}) - \sum_{k=1}^K \max(n_k^S, n_k^{\bar S}) \\ &\leqslant \sum_{k=1}^K \min(n_k^S, n_k^{\bar S}) - \sum_{k=1}^K \frac{|\Omega_i|}{2} = \sum_{k=1}^K \min(n_k^S, n_k^{\bar S}) - \frac{|\Omega|}{2} \; . \end{aligned}$$

Graph construction for E_{L_1}

$$E_{L_1}(\theta^S,\theta^{\bar S}) \equiv \sum_{k=1}^K \min(n_k^S, n_k^{\bar S}) - \frac{|\Omega|}{2} \; .$$

We add K *auxiliary nodes* $A_1, A_2, \dots, A_K$ into the graph and connect k^{th} auxiliary node to all the pixels that belong to the k^{th} bin. The capacity of all these links is set to $\beta = 1$.



Assume that bin k is split into n_k^S foreground and $n_k^{\bar S}$ background pixels. Then any cut separating the foreground and background must cut either n_k^S or $n_k^{\bar S}$ number of links that connect the pixels in bin k to the auxiliary node A_k. Therefore the *optimal cut* must choose $\min(n_k^S, n_k^{\bar S})$.

Binary segmentation with bounding box

Assume that a user provides a *bounding box* $R \subseteq \Omega$ around an object of interest and the goal is to perform *binary image segmentation* within the box. (Outside the bounding box are assigned to the background.)

The segmentation energy function is given by

$$E(S) = |\bar S \cap R| - \beta\|\theta^S - \theta^{\bar S}\|_{L_1} + \lambda|\partial S| \; ,$$

where the first term is a *standard ballooning* inside R, the second term is the *appearance overlap*, and the last term is a *contrast-sensitive smoothness* term.

$$|\partial S| = \sum_{(i,j)\in\mathcal{N}} w_{ij}|s_i - s_j| = \sum_{(i,j)\in\mathcal{N}} \frac{1}{\|i-j\|}\exp\left(-\frac{\|I(i)-I(j)\|^2}{2\sigma^2}\right)|s_i - s_j| \; .$$

σ_2 set as average value of $\|I(i) - I(j)\|^2$ over the image.

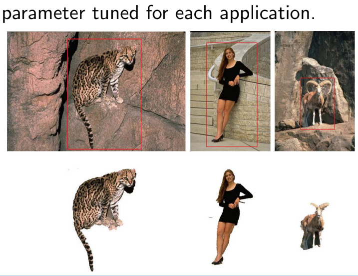
Note that this energy can be optimized with one graph cut.

Binary segmentation with bounding box

The image specific parameter β_{img} is adaptively set an based on the information within the provided bounding box:

$$\beta_{\text{img}} = \frac{|R|}{-\|\theta^R - \theta^{\bar R}\|_{L_1} + |\Omega|/2} \cdot \beta' \; ,$$

where β' is a global parameter tuned for each application.



Interactive segmentation with seeds

By making use of *seeds*, volumetric balancing (i.e. the term $h_\Omega(S)$) becomes unnecessary due to hard constraints enforced by the user.

Therefore, the segmentation energy is quite simple:

$$E_{\text{seeds}} = -\beta\|\theta^S - \theta^{\bar{S}}\|_{L_1} + \lambda|\partial S|$$

subject to the hard constraints given by the sees.

Template shape matching

Assume that we are given a *binary template mask* M, i.e.**shape prior**, and consider the combination of *shape matching cue* and *contrast sensitive smoothness* term via energy

$$E_1(S) = \min_{\rho\in\Phi} E_{\text{shape}}(S, M^\rho) + \lambda|\partial S| \ ,$$

where ρ denotes a transformation in parameter space Φ and M^ρ is a transformed binary mask.

The term $E_{\text{shape}}(S, M^\rho)$ measures the similarity (e.g., Hamming distance, L_2 distance) between segment S and the transformed binary mask M^ρ.

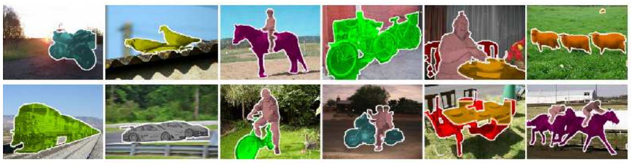
Template shape matching

Finally, the *appearance overlap* is also incorporated into the energy:

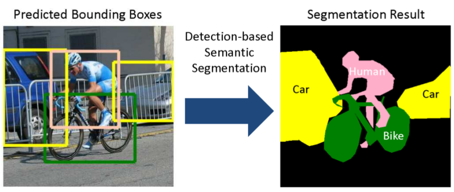
$$\begin{aligned} E_2(S) =& E_1(S) - \beta\|\theta^S - \theta^{\bar{S}}\|_{L_1} \\ =& \min_{\rho\in\Phi} E_{\text{shape}}(S, M^\rho) - \beta\|\theta^S - \theta^{\bar{S}}\|_{L_1} + \lambda|\partial S| \ . \end{aligned}$$

Segmentation with bounding box prior

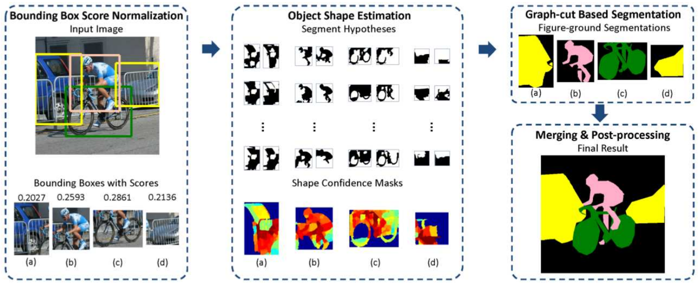
Semantic image segmentation



The motivation of our approach is based on some previous **detection-based approaches**.



Overview of the method



Constrained parametric min cut (CPMC)

Consider the binary image segmentation problem, where we are given foreground and background *seeds*. The foreground seeds, denoted by $\mathcal{V}_f$, are located on a grid of pixels, and the background seeds, denoted by $\mathcal{V}_b$, are set along the image border.

We compute the figure-ground segmentations resulting from minimum cuts respecting a seed hypothesis for several values of the foreground bias, including negative ones.

Assuming an undirected graphical model $G = (\mathcal{V}, \mathcal{E})$ with the corresponding energy:

$$E(x) = \sum_{i\in\mathcal{V}} E_i^\lambda(x_i) + \sum_{(i,j)\in\mathcal{E}} [\![x_i \neq x_j]\!] E_{ij}(x_i, x_j) \ ,$$

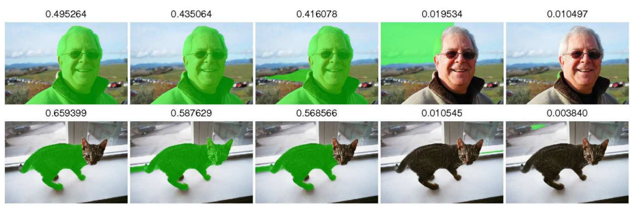
where E_i^λ is the data term for $\lambda \in \mathbb{R}$, and E_{ij} is a contrast sensitive term based on the response of egde detection.

CPMC: Data term

The data term for $\lambda \in \mathbb{R}$ is defined as:

$$E_i^\lambda(x_i) = \begin{cases} 0, & \text{if } x_i = 1, i \notin \mathcal{V}_b \\ \infty, & \text{if } x_i = 1, i \in \mathcal{V}_b \\ \infty, & \text{if } x_i = 0, i \in \mathcal{V}_f \\ \log\frac{p_f(x_i)}{p_b(x_i)} + \lambda, & \text{if } x_i = 0, i \notin \mathcal{V}_f \end{cases}$$

Some resulting segments for different values of λ:

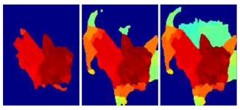


Object shape estimation

- Generate a pool of segments via CPMC (without any ranking procedure)
- Calculate the average score map from the obtained segments

$$\bar{M}(\mathbf{p}) = \frac{1}{k}\sum_{i=1}^{k} \mathbb{1}_i(\mathbf{p}) \,.$$

- Threshold the average map at different t



$$\mathcal{M}_t = \{\mathbf{p} \in \mathbb{R}^2 | \bar{M}(\mathbf{p}) \geqslant t\} \,.$$

- Select the best overlapping segments ($\mathcal{S}_{i^*}$)with the object boundary

$$i^* = \underset{i \in \{1,\dots,k\}}{\operatorname{argmax}} \left\{ \max_{t \geqslant \mu \max(\bar{M})} \frac{|\mathcal{M}_t \cap \mathcal{S}_i|}{|\mathcal{M}_t \cup \mathcal{S}_i|} \right\} \,.$$

Object shape estimation

- The shape guidance is based on the best segments

$$M(\mathbf{p}) = \bar{M}(\mathbf{p}) \mathbb{1}_{i^*}(\mathbf{p}) \,.$$

Some exemplar estimated object shape:

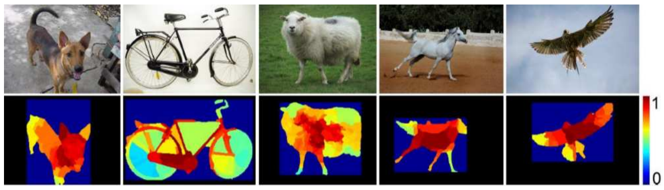


Figure-ground segmentation

We define a simple energy over **super-pixels**:

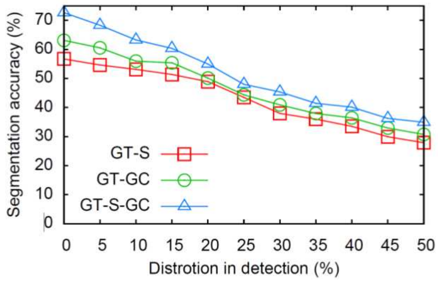
$$E(\mathbf{x}) = \sum_{i \in \mathcal{V}} u_i(x_i) + \sum_{(i,j) \in \mathcal{E}} v_{ij}(x_i, x_j) \,.$$

- Pairwise term v_{ij} is the contrast-sensitive Potts-model
- Data term u_i is based on appearance and shape terms

$$u_i(x_i) = -\alpha \log \underbrace{\left(A(x_i)\right)}_{\textbf{Appearance term}} - (1-\alpha) \log \underbrace{\left(S(x_i)\right)}_{\textbf{Shape term}} \,.$$

$S(x_i = 1)$ is calculated based on the average value of M over the given super-pixel, and $S(x_i = 0) := 1 - S(x_i = 1)$.

Effect of detection



- GT-S: shape guidance without graph-cut
- GT-GC: graph-cut without shape guidance
- GT-S-GC: graph-cut with shape guidance

Summary

A detection-based approach has been proposed for semantic segmentation:

- by applying a simple voting scheme, based on a generated pool of segment hypotheses, shape guidance is estimated
- there is no need for training data

- it heavily relies on object detection
  (the performance might improve if better detection method is available)
- a better way to handle multiple interacting objects in the merging step should be considered.

Literature *

- Meng Tang, Lena Gorelick, Olga Veksler, and Yuri Boykov. **GrabCut in One Cut**. In *Proceedings of IEEE International Conference on Computer Vision*, pp. 1769–1776, Sydney, Australia, December, 2013.
- Wei Xia, Csaba Domokos, Jian Dong, Loong-Fah Cheong, and Shuicheng Yan. **Semantic Segmentation without Annotating Segments**. In *Proceedings of International Conference on Computer Vision*, pp. 2176–2183, Sydney, Australia, December, 2013.