GrabCut in One Cut

Meng Tang Lena Gorelick Olga Veksler Yuri Boykov
University of Western Ontario
Canada

Abstract

Among image segmentation algorithms there are two major groups: (a) methods assuming known appearance models and (b) methods estimating appearance models jointly with segmentation. Typically, the first group optimizes appearance log-likelihoods in combination with some spacial regularization. This problem is relatively simple and many methods guarantee globally optimal results. The second group treats model parameters as additional variables transforming simple segmentation energies into high-order NP-hard functionals (Zhu-Yuille, Chan-Vese, Grab-Cut, etc). It is known that such methods indirectly minimize the appearance overlap between the segments.

We propose a new energy term explicitly measuring L_1 distance between the object and background appearance models that can be globally maximized in one graph cut. We show that in many applications our simple term makes NP-hard segmentation functionals unnecessary. Our one cut *algorithm effectively replaces approximate iterative optimization techniques based on block coordinate descent.*

1. Introduction

Appearance models are critical for many image segmentation algorithms. The most basic object segmentation energy [3, 20] combines boundary regularization with log-likelihood ratios for fixed foreground and background appearance models, *e.g.* color distributions, θ^1 and θ^0

$$E(S|\theta^1,\theta^0) \;=\; -\sum_{p\in\Omega} \ln \Pr(I_p|\theta^{s_p}) \;+\; |\partial S| \tag{1}$$

where Ω is the set of all image pixels. Commonly used length regularization is $|\partial S| = \sum_{\{p,q\}\in\mathcal{N}} \omega_{pq}|s_p - s_q|$ where $s_p \in \{0,1\}$ are binary indicator variables for segment $S \subset \Omega$ and $\mathcal{N}$ is the set of all pairs of neighboring pixels. One important practical advantage of this basic energy is that there are efficient methods for their global minimization using graph cuts [4] or continuous relaxations [5, 18].

In many applications the appearance models may not be known *a priori*. Some well-known approaches to segmentation [25, 19, 6] consider model parameters as extra optimization variables in their segmentation energies. E.g.,

$$E(S,\theta^1,\theta^0) \;=\; -\sum_{p\in\Omega} \ln \Pr(I_p|\theta^{s_p}) \;+\; |\partial S|, \tag{2}$$

which is known to be NP-hard [22], is used for interactive segmentation in GrabCut [19] where initial appearance models θ^1, θ^0 are computed from a given bounding box. The most common approximation technique for minimizing (2) is a block-coordinate descent [19] alternating the following two steps. First, they fix model parameters θ^1, θ^0 and optimize over S, e.g. using a graph cut algorithm for energy (1) as in [3]. Second, they fix segmentation S and then optimize over model parameters θ^1 and θ^0. Two well-known alternatives, *dual decomposition* [22] and *branch-and-mincut* [14], can find a global minimum of energy (2), but these methods are too slow in practice.

We observe that when appearance models θ^1, θ^0 are represented by (non-parametric) color histograms, minimization of (2) is equivalent to minimization of energy

$$E(S) \;=\; |S|\cdot H(\theta^S) + |\bar{S}|\cdot H(\theta^{\bar{S}}) + |\partial S| \tag{3}$$

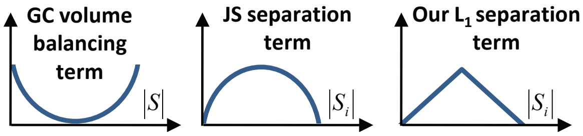
that depends on S only. Here θ^S and $\theta^{\bar{S}}$ are histograms inside object S and background $\bar{S} = \Omega \setminus S$, and $H(\cdot)$ is the *entropy* functional for probability distributions. This form of energy (2) can be obtained by replacing the sum over pixels in (2) by the sum over color bins. Well-known inequalities for *cross entropy*, e.g. $H(\theta^S|\theta^1) \geq H(\theta^S)$, also help. Interestingly, the global minimum of segmentation energy (3) does not depend on the initial color models provided by the user. Thus, the interactivity of GrabCut algorithm is primarily due to the fact that its solution is a local minimum of (3) sensitive to the initial bounding box.

Formulation (3) is useful for analyzing the properties of energy (2). The entropy terms of this energy prefer segments with more peaked color distributions. Intuitively, this should also imply distributions with small overlap. For example, consider a simple case of black-&-white images when color histograms θ^1 and θ^0 have only two bins. Clearly, the lowest value (zero) for the entropy terms in (3) is achieved when black and white pixels are completely separated between the segments, e.g. all white pixels are inside the object and all black pixels are inside the background.

In general, the color separation bias in energy (3) is shown by equivalently rewriting its two entropy terms as

$$h_\Omega(S) - \sum_i h_{\Omega_i}(S_i) \tag{4}$$

where $h_A(B) = |B| \cdot \ln|B| + |A \setminus B| \cdot \ln|A \setminus B|$ is standard Jensen-Shannon (JS) divergence functional for subset $B \subset A$. We also use Ω_i to denote the set of all pixels in color bin i (note $\Omega = \cup_i \Omega_i$) and $S_i = S \cap \Omega_i$ is a subset of pixels of color i inside object segment (note $S = \cup_i S_i$). The plots for functions $h_\Omega(S)$ and $-h_{\Omega_i}(S_i)$ are illustrated in Fig.1. The first term in (4) shows that energies (2) or (3) implicitly bias image segmentation to two parts of equal size, see Fig.1(a). The remaining terms in (4) show bias to color separation between the segments, see Fig.1(b). Note that a similar analysis in [22] is used to motivate their convex-concave approximation algorithm for energy (2).



(a) $h_\Omega(S)$ (b) $-h_{\Omega_i}(S_i)$ (c) $-|S_i - \bar{S}_i|$

Figure 1. Energy (3): volume balancing (a) and Jensen-Shannon color separation terms (b). Our L_1 color separation term (c).

Volume balancing $h_\Omega(S)$ is the only term in (4) and (2) that is not submodular and makes optimization difficult. Our observation is that in many applications this volume balancing term is simply unnecessary, see Sections 3.1.3, 3.2-3.3. In other applications we propose to replace it by other easier to optimize terms.

Moreover, while it is known that JS color separation term $-h_{\Omega_i}(S_i)$ is submodular[1] and can be optimized by graph cuts [11, 12, 22], we propose to replace it with a more basic L_1 separation term in Fig.1(c). We show that it corresponds to a simpler construction with much fewer auxiliary nodes leading to higher efficiency. Interestingly, it also gives better color separation effect, see Section 3.1.2.

We also observe one practical limitation of block-coordinate approach to (2), as in GrabCut [19], could be due to deteriorating sensitivity to local minima when the number of color bins for models θ^S and $\theta^{\bar{S}}$ is increased, see Table 2 and Fig.5. In practice, however, finer bins better capture the information contained in the full dynamic range of color images (8-bit per channel or more). Our ROC curves show that even a difficult camouflage image in Figure 2 has a good separation of intensities between the object and background if larger number of bins is used. With 16^3 bins, however, the overlap between the "fish" and the background is too strong making it hard to segment. Since GrabCuts algorithm is more likely to get stuck at weak local minima for larger number of bins, it may not benefit from higher color resolution, see Table 2 and Fig.5.

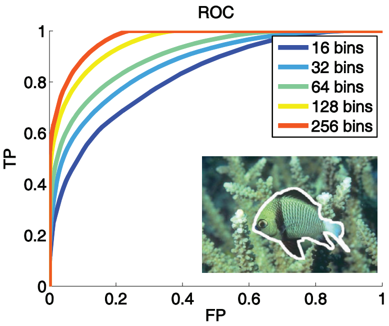


Figure 2. Given appearance models θ^1, θ^0 extracted from the ground truth object/background segments, we can threshold log-likelihood ratios $\ln \frac{\theta^1(I_p)}{\theta^0(I_p)}$ at each pixel p and compare the result with the same ground truth segmentation. The corresponding ROC curves show that the color separation between the object and background increases for finer bins.

Our contributions are summarized below:

- We propose a simple energy term penalizing L_1 measure of appearance overlap between segments. While it can be seen as a special case of a high-order *label consistency* term introduced by Kohli et al. [11, 12] we propose a simpler construction for our specific constraint. Unlike NP-hard multi-label problems discussed in [11, 12], we focus on binary segmentation where such high-order constraints can be globally minimized. Moreover, we show that our L_1 term works better for separating colors than other concave separators (including JS, Bhattacharyya, and χ^2).

- We are first to demonstrate fast globally optimal binary segmentation technique explicitly minimizing overlap between object/background color distributions. In one graph cut we get similar or better results at faster running times w.r.t. earlier methods, e.g. [19, 22, 14, 7].

- We show general usefulness of the proposed appearance overlap penalty by showing different practical applications: binary segmentation, shape matching, etc.

2. Minimizing appearance overlap in One-Cut

Let $S \subset \Omega$ be a segment and denote by θ^S and $\theta^{\bar{S}}$ the unnormalized color histograms for the foreground and background appearance respectively. Let n_k be the number

[1] Any concave function of cardinality is submodular [16]. This applies to JS, χ^2, Bhattacharyya, and our L_1 color separation terms in Figs.1, 9.

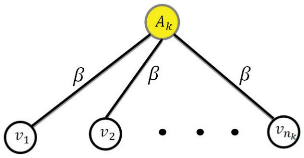


Figure 3. Graph construction for E_{L_1}: nodes $v_1, v_2, \ldots, v_{n_k}$ corresponding to the pixels in bin k are connected to the auxiliary node A_k using undirected links. The capacity of these links is the weight of appearance overlap term $\beta > 0$.

of pixels in the image the belong to bin k and let n_k^S and $n_k^{\bar{S}}$ be the according number of the foreground and background pixels in bin k. Our appearance overlap term penalizes the intersection between the foreground and background bin counts by incorporating the simple yet effective high-order L_1 term into the energy function:

$$E_{L_1}(\theta^S, \theta^{\bar{S}}) = -\|\theta^S - \theta^{\bar{S}}\|_{L_1}. \tag{5}$$

Below we explain how to incorporate and optimize the term $E_{L_1}(\theta^S, \theta^{\bar{S}})$ using one graph cut. For the clarity of the explanation we rewrite

$$E_{L_1}(\theta^S, \theta^{\bar{S}}) = \sum_{k=1}^{K} \min(n_k^S, n_k^{\bar{S}}) - \frac{|\Omega|}{2}. \tag{6}$$

It is easy to show that the two sides of (6) are equivalent. The details of the graph construction for the above term are shown in Fig. 3.

We add K auxiliary nodes $A_1, A_2, ..., A_K$ into the graph and connect k^{th} auxiliary node to all the pixels that belong to the k^{th} bin. In this way each pixel is connected to its corresponding auxiliary node. The capacity of these links is set to $\beta = 1$. Assume that bin k is split into foreground and background, resulting in n_k^S and $n_k^{\bar{S}}$ pixels accordingly. Then any cut separating the foreground and background pixels must either cut n_k^S or $n_k^{\bar{S}}$ number of links that connect the pixels in bin k to the auxiliary node A_k. The optimal cut must choose $\min(n_k^S, n_k^{\bar{S}})$ in (6).

A similar graph construction with auxiliary nodes is proposed in [11, 12] to minimize higher order pseudo-boolean functions of the following form:

$$f(X_c) = \min\{\theta_0 + \sum_{i \in c} w_i^0 (1 - x_i), \theta_1 + \sum_{i \in c} w_i^1 x_i, \theta_{max}\} \tag{7}$$

where $x_i \in \{0, 1\}$ are binary variables in clique c, $w_i^0 \geqslant 0$, $w_i^1 \geqslant 0$, and θ_0, θ_1 and θ_{max} are parameters satisfying the constraints $\theta_{max} \geqslant \theta_0$ and $\theta_{max} \geqslant \theta_1$.

Below we discuss the relation and the differences between the two constructions. The construction in [11, 12] can be used to minimize E_{L_1} as follows: consider each color bin as a clique and set parameters $w_i^0 = 1$, $w_i^1 = 1$, $\theta_0 = 0$, $\theta_1 = 0$ and $\theta_{max} = n_k/2$, where n_k is the number of pixels in bin k. Then $f(X_c)$ reduces to $E_{L_1}(\theta^S, \theta^{\bar{S}}) + |\Omega|/2$. One advantage of our graph construction is that we only need one auxiliary node for each bin in contrast to two auxiliary nodes in [11, 12]. Furthermore, our construction also extends to pseudo-boolean functions in (7) using directed links, as shown in Fig. 4.

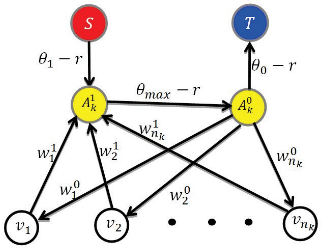


Figure 4. Graph construction for minimizing pseudo-boolean function (7), $r = \min\{\theta_0, \theta_1\}$.

To see how it works we consider four possible label assignments for the auxiliary nodes A_k^1 and A_k^0. The table below shows the cost of corresponding cuts. The minimum cut renders optimization of the function (7).

(A_k^1, A_k^0)	the cost of cut
(0,0)	$\theta_1 + \sum_{i \vert x_i = 1} w_i^1 - r$
(0,1)	$\theta_0 + \sum_{i \vert x_i = 0} w_i^0 + \theta_1 + \sum_{i \vert x_i = 1} w_i^1 - 2r$
(1,0)	$\theta_{max} - r$
(1,1)	$\theta_0 + \sum_{i \vert x_i = 0} w_i^0 - r$

Table 1. Cut costs corresponding to four possible label assignments to the binary auxiliary nodes A_k^1 and A_k^0. The optimal cut must choose the minimum of the above costs, thus minimizing (7).

Unlike our construction, the method in [11, 12] requires that $\forall X_c \in \{0, 1\}^{|c|}$ the parameter θ_{max} in $f(X_c)$ should satisfy the following constraint:

$$\theta_{max} \leqslant \max(\theta_0 + \sum_{i \in c} w_i^0 (1 - x_i), \theta_1 + \sum_{i \in c} w_i^1 x_i). \tag{8}$$

In contrast, we can optimize high-order functions in (7) with arbitrary θ_{max}, provided that $\theta_{max} \geqslant \theta_0$ and $\theta_{max} \geqslant \theta_1$.

3. Applications

In this section we apply our appearance overlap penalty term in a number of different practical applications including interactive binary segmentation in Sec.3.1, shape matching in Sec.3.2, and saliency detection in Sec.3.3.

3.1. Interactive segmentation

First, we discuss interactive segmentation with several standard user interfaces: bounding box [19] in Section 3.1.1 and seeds [3] in Section 3.1.3. We compare different color separation terms in Section 3.1.2.

	Error rate	Mean runtime
GrabCut (8^3 bins)	8.54%	2.48 s
GrabCut (16^3 bins)	7.1%[2]	1.78 s
GrabCut (32^3 bins)	8.78%	1.63s
GrabCut (64^3 bins)	9.31%	1.64s
GrabCut (128^3 bins)	11.34%	1.45s
GrabCut (256^3 bins)	13.59%	1.46s
DD (16^3 bins)	10.5%	576 s
One-Cut (8^3 bins)	9.98%	18 s
One-Cut (16^3 bins)	8.1%	5.8 s
One-Cut (32^3 bins)	6.99%	2.4 s
One-Cut (64^3 bins)	6.67%	1.3 s
One-Cut (128^3 bins)	6.71%	0.8 s
One-Cut (256^3 bins)	7.14%	0.8 s

Table 2. Error rates and mean runtime for GrabCut [19], Dual Decomposition (DD) [22], and our method, denoted by *One-Cut*.

3.1.1 Binary segmentation with bounding box

First, we use appearance overlap penalty in a binary segmentation experiment *a la* GrabCut [19]. A user provides a bounding box around an object of interest and the goal is to perform binary image segmentation within the box. The pixels outside the bounding box are assigned to the background using hard constraints. Let $R \subseteq \Omega$ denote the binary mask corresponding to the bounding box, $S_{GT} \subseteq \Omega$ be the ground truth segmentation and $S \subseteq \Omega$ be a segment. Denote by $1_S = \{s_p | p \in \Omega\}$ the characteristic function of S. The segmentation energy function $E(S)$ is given by

$$E(S) = |\bar{S} \cap R| - \beta \|\theta^S - \theta^{\bar{S}}\|_{L_1} + \lambda |\partial S|, \qquad (9)$$

where the first term is a standard ballooning inside the bounding box R, the second term is the appearance overlap penalty as in (5), and the last term is a contrast-sensitive smoothness term. We use $|\partial S| = \sum \omega_{pq} |s_p - s_q|$ with $\omega_{pq} = \frac{1}{\|p-q\|} \cdot e^{\frac{-\Delta I^2}{2\sigma^2}}$ and σ^2 set as average ΔI^2 over the image. This energy can be optimized with one graph cut.

It is common to tune the relative weight of each energy term for a given dataset [22]. The input bounding box contains useful information about the object to be segmented. We use the measure of appearance overlap between the box R and its background $\bar{R}$ to automatically find image specific relative weight β of the appearance overlap term w.r.t. the first (ballooning) term in (9). In our experiments, we adaptively set an image specific parameter β_{Img} based on the information within the provided bounding box:

$$\beta_{Img} = \frac{|R|}{-\|\theta^R - \theta^{\bar{R}}\|_{L_1} + |\Omega|/2} \cdot \beta'. \qquad (10)$$

Here β' is a global parameter tuned for each application. We found it to be more robust compared to tuning β.

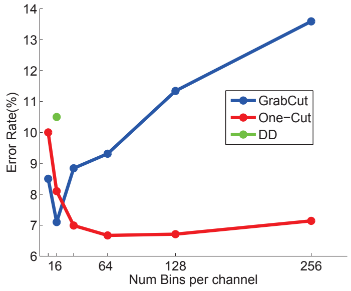


Figure 5. Error-rates for different bin resolutions, as in Table 2.

We experiment on the well known Grab-cut database [19][3]. The error rate is defined as the number of misclassified pixels within the bounding box R divided by the size of the box $|R|$. We test with different number of color bins and provide quantitative comparison with the grab-cut method [19] (our implementation, modified to work with histograms as in [22]) and the dual decomposition method [22] (results reported by the authors). The Table 2 and the plots in Fig. 5 report the respective error rates and the average running times. We tune λ separately for each method and number of bins.

With 16^3 bins, the GrabCut method is the most accurate and fast. However, it is important to see the effect of working with larger number of bins, as some objects might only be distinguishable from the background when using higher dynamic rage. As we increase the number of color bins, the error rate for the GrabCut method increases, while the error rate of One-Cut decreases. When using 128^3 bins One-Cut runs twice as fast, while obtaining much lower error rate. This is because with more bins, more auxiliary nodes are used, but each auxiliary node is connected to less pixels. The connectivity density decreases and the mincut/maxflow algorithm runs faster. DD is hundreds of times slower, while its error rate is quite high. Note that in [22], images are down-scaled to maximum side-length of 250 pixels while the method here deals with the original image.

Finally, Figures 6-7 show examples of input bounding boxes and segmentation results with the GrabCut [19], Dual Decomposition [22] and our One-Cut method.

3.1.2 Comparison of Appearance Overlap Terms

Below we discus additional variants for appearance overlap penalty term. We explain how they all can be implemented with the construction proposed in Fig. 4 and compare their performance in the task of binary segmentation applied to the GrabCut dataset [19]. We consider four appearance overlap terms based on the L_1 norm, χ^2 distance, Bhattacharyya coefficient and Jensen-Shannon divergence

[2] The error rate for our implementation of the GrabCut method is slightly different from 8.1% reported in [22], since we use a different smoothness term and do not downscale images.

[3] There are 50 images in the dataset, but we exclude the cross image for the sake of comparison with [22].

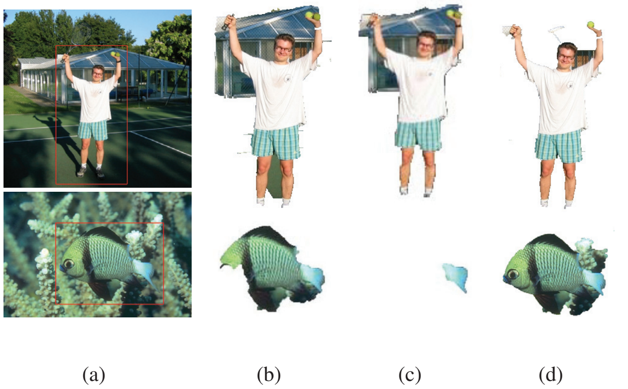


Figure 6. Example of segmentation results. From left to right: (a) user input, (b) GrabCut [19], (c) Dual Decomposition (DD) [22], (d) our One-Cut. For these examples we used 16^3 bins.

between histograms. To that end we define

$$D_{L_1}(\theta^S,\theta^{\bar{S}}) = \sum_{k=1}^{K} \min\left(n_k^S, n_k^{\bar{S}}\right) \tag{11}$$

$$D_{\chi^2}(\theta^S,\theta^{\bar{S}}) = \sum_{k=1}^{K} n_k/2 - (n_k^S - n_k^{\bar{S}})^2/(2n_k) \tag{12}$$

$$D_{Bha}(\theta^S,\theta^{\bar{S}}) = \sum_{k=1}^{K} \sqrt{n_k^S n_k^{\bar{S}}} \tag{13}$$

$$D_{JS}(\theta^S,\theta^{\bar{S}}) = \sum_{k=1}^{K} \frac{n_k \log n_k - n_k^S \log n_k^S - n_k^{\bar{S}} \log n_k^{\bar{S}}}{2} \tag{14}$$

where θ^S and $\theta^{\bar{S}}$ are unnormalized histograms of the foreground and background respectively. The D_{L_1} term above is equivalent to $-\|\theta^S - \theta^{\bar{S}}\|_{L_1}$, but we use this notation for easiness of comparison with other overlap terms. All four terms above are concave functions of n_k^s attaining maximum at $n_k/2$. See Fig. 9 (top-left) for the visualization of the terms and comparison with D_{L_1}.

Similarly to [11] we observe that any concave function can be approximated as a piece-wise linear function by using a summation of specific (pyramid-like) truncated functions, each having a general form as in (7). For example, Fig. 8 illustrates one possible approximation using three components. These truncated components can be incorporated into our graph using the construction shown in Fig. 4. Note, D_{L_1} is equivalent to D_{χ^2}, D_{Bha} or D_{JS} when approximated using one truncated component.

All three appearance overlap terms above can be optimized with one graph cut. We wish to find out which term and what level of approximation accuracy are optimal for the task of binary segmentation. Therefore, for each term we vary the approximation accuracy (the number of truncated components used) and compare the performance of D_{χ^2}, D_{Bha}, D_{JS} with that of D_{L_1}.

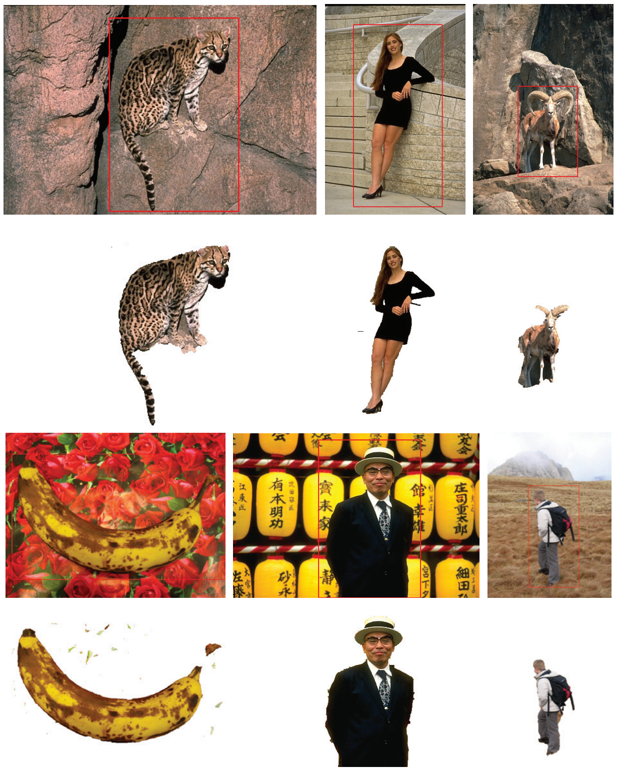


Figure 7. Example of segmentation results obtained with our One-Cut. For these examples we used 128^3 bins.

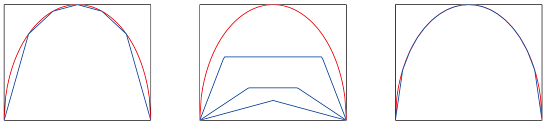


Figure 8. The original concave function (red) is approximated as a piece-wise linear function (blue, left) using three truncated components (blue,middle). Approximation with ten components (blue, right) is already very accurate.

In the first experiment, we use an adaptive image specific weight β_{Img} for the appearance overlap term as in (10) and set $\beta^{'} = 0.9$ which was found optimal for D_{L_1} overlap term. Fig. 9 (top-right) shows that as the approximation accuracy (the number of components used) increases, the error rate goes up.

In the second experiment, we choose the optimal β_{Img} separately for each appearance overlap term by replacing the denominator of (10) with either $D_{\chi^2}(\theta^R,\theta^{\bar{R}})$, $D_{Bha}(\theta^R,\theta^{\bar{R}})$ or $D_{JS}(\theta^R,\theta^{\bar{R}})$ according to the appearance overlap term used. We also tune parameter $\beta^{'}$ separately for each appearance overlap term. As shown in Fig. 9 (bottom-right), D_{L_1} achieves the lowest error rate and has the shortest running time (bottom-right) than any other

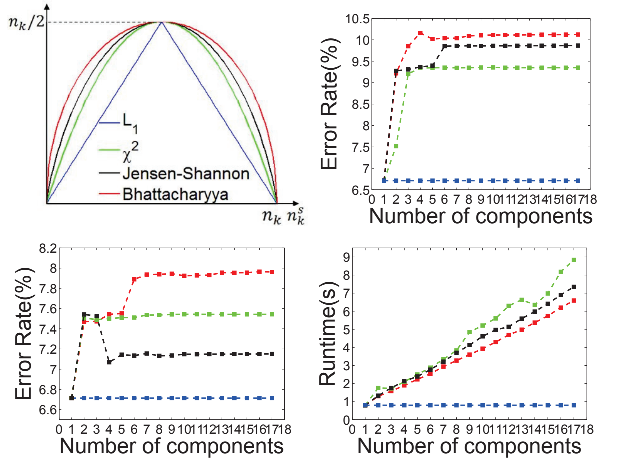


Figure 9. Comparison of appearance overlap terms: (top-left) shows the functions plotted for one bin k, (top-right) shows segmentation error rates using the same β_{Img} as in (10) for all overlap terms and (bottom-left) shows segmentation results when using a term-specific β_{Img}. The running time is shown on (bottom-right).

overlap term with any level of approximation accuracy.

In the third experiment we replace D_{L_1} with the truncated version $D_{L_1^T} = \sum_{k=1}^{K} \min(n_k^S, n_k^{\bar{S}}, t \cdot n_k/2)$ where $t \in [0, 1]$ is the truncation parameter. Our D_{L_1} term can be seen as a special case of the truncated $D_{L_1^T}$ where $t = 1$. Again, for each value of t we replace the denominator in (10) by $D_{L_1^T}(\theta^R, \theta^{\bar{R}})$ and tune β'. Fig. 10 reports the error rates of the segmentation as a function of the parameter t. It can be seen that the non-truncated version ($t = 1$) yields the best performance. This further proves the benefit of our proposed D_{L_1} appearance overlap term.

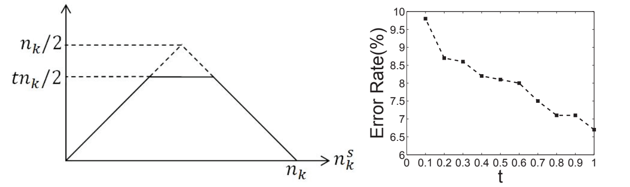


Figure 10. Left: Truncated appearance overlap term $D_{L_1^T}$ for a bin k. Right: Segmentation error rate as a function of parameter t in $D^T_{L_1}$. Best results are achieved for $t = 1$ (no truncation).

3.1.3 Interactive Segmentation with Seeds

Unlike interactive segmentation with abounding box, using seeds *a la* Boykov-Jolly [3] makes volumetric balancing unnecessary due to hard constraints enforced by the user. Therefore, the segmentation energy is quite simple:

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

subject to the hard constraints given by the sees. Figure 11 shows several qualitative segmentation results.

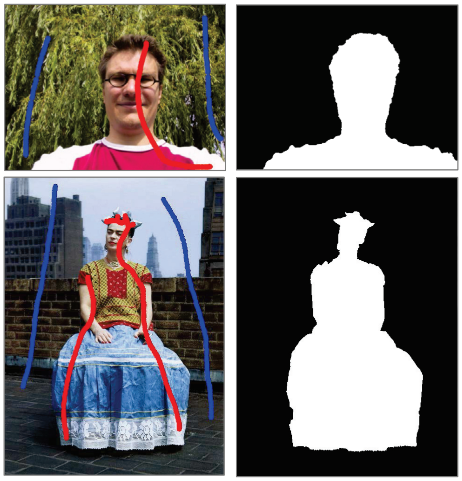


Figure 11. Interactive segmentation with seeds

3.2. Template Shape Matching

Below we discuss how appearance overlap penalty term can be used for template shape matching. Several prior methods rely on graph-cut based segmentation with shape prior [21, 8, 14, 23]. Most commonly, these methods use a binary template mask M and combine the shape matching cue a contrast sensitive smoothness term via energy

$$E_1(S) = \min_{\rho \in \Phi} E_{Shape}(S, M^\rho) + \lambda |\partial S|. \tag{15}$$

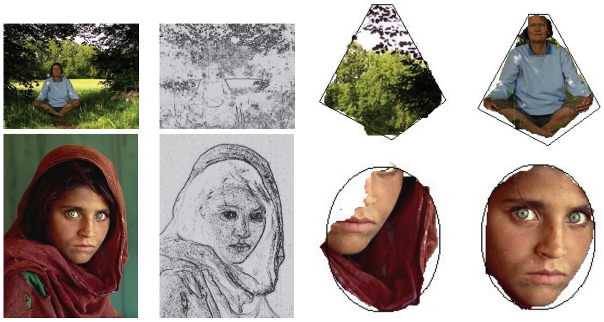
where ρ denotes a transformation in parameter space Φ and M^ρ is a transformed binary mask. The term $E_{Shape}(S, M^\rho)$ measures the similarity between segment S and the transformed binary mask M^ρ. Possible metric include Hamming distance or L_2 distance. We further incorporate the appearance overlap into the energy:

$$E_2(S) = E_1(S) - \beta \|\theta^S - \theta^{\bar{S}}\|_{L_1} \tag{16}$$

and compare the performance of $E_1(S)$ and $E_2(S)$ in the task of template shape matching. Fig. 12 shows few examples of input template and matching results. Without the appearance overlap term shape matching yields erroneous segmentation due to the clutter edges in the background.

We experiment on Microsoft Research Cambridge Object Recognition Image Database[4]. There are 282 side view images of cars, roughly of the same scale. We down-scaled all images to 320×240 and used 128^3 color bins per image. For this experiment, Φ to defined be the set of all possible translations and horizontal flip. For the template matching, we scan the image in a sliding-window fashion and compute maxflow/mincut with dynamic graph cut [13]. We use Hamming distance in (15). In principle, branch-and-mincut [14] can speed up optimization of both energies (15) and (16), but this is outside the scope of our paper.

[4] http://research.microsoft.com/en-us/projects/objectclassrecognition



(a) (b) (c) (d)

Figure 12. Template shape matching examples: (a) Original images, (b) Contrast sensitive edge weights, (c-d) Shape matching with and without the appearance overlap penalty. Input shape templates are shown as contours around the resulting segmentation.

Fig. 13 shows the coarse car template used for this experiment and some qualitative results. Table 3 provides quantitative comparison of the results obtained with and without incorporating the appearance overlap term, namely using $E_2(S)$ and $E_1(S)$. The results are reported with respect to manually outlined ground truth segmentations and clearly point to the benefit of incorporating the overlap term E_{L_1} into the segmentation energy. When using $E_2(S)$ we achieve higher true positive (TP) rate of 81.88%, lower false positive (FP) rate of 3.86% and less misclassified pixels without compromising much the running time.

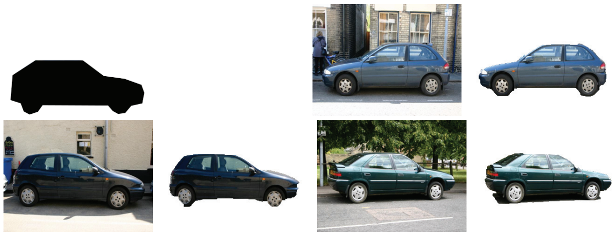


Figure 13. Template shape matching examples: shape (top left) and pairs of original images + segmentations with $E_2(S)$.

Energy	TP	FP	Error pixels	Runtime
$E_1(S)$	76.97%	6.96%	10106	4.1 s
$E_2(S)$	81.88%	3.86%	7480	13.0 s

Table 3. Template shape matching: comparing $E_1(S)$ and $E_2(S)$ in terms of TP, FP, misclassified pixels, and mean running time. We used $\lambda = 5$ for $E_1(S)$ and $(\lambda = 5, \beta = 1.1)$ for $E_2(S)$.

3.3. Salient object segmentation

Salient region detection and segmentation is an important preprocessing step for object recognition and adaptive compression. Salient objects usually have an appearance that is distinct from the background [1, 7, 17]. Below we show how our appearance overlap penalty term can be used for the task of salient object segmentation. We use the saliency map provided by [17] because it yields the best precision/recall curve when thresholded and compared to the ground truth. Let $A : \Omega \rightarrow [0, 1]$ denote the normalized saliency map and $<A>$ be its mean value. Then let

$$E_{Salience}(S) = \sum_{p\in\Omega} <A> -(A(p)) \cdot s_p \qquad (17)$$

denote an energy term measuring the saliency of a given segment. We now define two segmentation energies, with and without the appearance overlap term. Let $E_3(S)$ be the energy combining the saliency and smoothness terms

$$E_3(S) = E_{Salience}(S) + |\partial S|, \qquad (18)$$

and $E_4(S)$ be the energy with the appearance overlap term

$$E_4(S) = E_3(S) - \beta \|\theta^S - \theta^{\bar{S}}\|_{L_1}. \qquad (19)$$

$E_4(S)$ can be optimized in one graph cut using the construction shown in Fig. 3. We use 128^3 color bins, $\beta = 0.3$ and smoothness term $\omega_{pq} = 3(e^{\frac{-\partial I^2}{2\sigma^2}} / ||p - q|| + 0.1)$.

We experiment on publicly available dataset [1] which provides ground-truth segmentation of 1000 images from MSRA salient object database [15]. Fig. 14 compares the performance of $E_3(S)$ and $E_4(S)$ with that of FT [1], CA [9], LC [24], HC [7] and RC [7] in terms of *precision*, *recall* and *F-measure* defined as

$$F = \frac{1.3 \cdot Precision \cdot Recall}{0.3 \cdot Precision + Recall}. \qquad (20)$$

Optimizing $E_3(S)$ results in *precision* = 91%, *recall* = 85% and *F-measure* = 86%, whereas incorporating the appearance term in $E_4(S)$ yields *precision* = 89%, *recall* = 89% and *F-measure* = 89%, which is comparable to the state-of-the-art results reported in literature [7](*precision* = 90%, *recall* = 90%, *F-measure* = 90%). Note that our optimization requires one graph-cut only, rather than the iterated EM-style grab-cut refinement in [7]. Assuming the saliency map is precomputed, the average running time for optimizing $E_4(S)$ is 0.43s and for optimizing $E_3(S)$ is 0.39s. Fig. 15 shows qualitative results for our saliency segmentation with and without the appearance overlap term.

4. Conclusions and Future work

We proposed an appearance overlap term for graph-cut based image segmentation. This term is based on L_1 distance between unnormalized histograms of foreground and background. We show that this term is easier to implement and that it is more effective at separating colors than compared to other concave (submodular) separators. While L_1 appearance overlap term is a special case of robust P^n-Potts

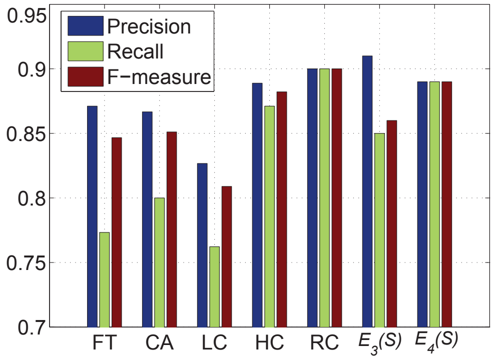


Figure 14. Saliency segmentation results reported for dataset [1]: Precison-Recall and F-measure bars for $E_3(S)$, $E_4(S)$ are compared to FT[1], CA[9], LC[24], HC[7] and RC[7].

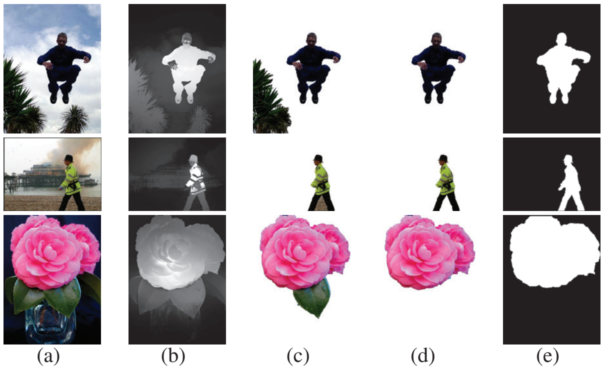


Figure 15. Saliency segmentation examples: (a) Original image, (b) Saliency map from [17] with bright intensity denoting high saliency, (c-d) Graph cut segmentation without and with appearance overlap penalty term, (e) Ground truth.

model, we show a simpler construction that can be easily incorporated into any graph cut based segmentation method. In several applications including interactive image segmentation, shape matching and saliency region detection we achieve the state-of-the-art results. We show that our term is a good fit for interactive segmentation (with bounding box or user seeds interfaces). In contrast to other appearance adaptive methods (e.g. GrabCut) our approach finds guaranteed global minimum in one cut.

Future work may include combining submodular L_1 color separation term with problematic non-submodular volume balancing terms like h_Ω in (4), which could be efficiently optimized using FTR [10] or *auxiliary cuts* [2].

References

[1] R. Achanta, S. Hemami, F. Estrada, and S. Süsstrunk. Frequency-tuned salient region detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. 7, 8

[2] I. B. Ayed, L. Gorelick, and Y. Boykov. Auxiliary cuts for gen. classes of higher order func. In *CVPR*, pages 1304–1311, 2013. 8

[3] Y. Boykov and M.-P. Jolly. *Interactive graph cuts* for optimal boundary & region segmentation of objects in N-D images. In *ICCV*, volume I, pages 105–112, July 2001. 1, 3, 6

[4] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 26(9):1124–1137, September 2004. 1

[5] T. Chan, S. Esedoglu, and M. Nikolova. Algorithms for finding global minimizers of image segmentation and denoising models. *SIAM Journal on Applied Mathematics*, 66(5):1632–1648, 2006. 1

[6] T. Chan and L. Vese. Active contours without edges. *IEEE Trans. Image Processing*, 10(2):266–277, 2001. 1

[7] M.-M. Cheng, G.-X. Zhang, N. J. Mitra, X. Huang, and S.-M. Hu. Global contrast based salient region detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011. 2, 7, 8

[8] D. Freedman and T. Zhang. Interactive graph cut based segmentation with shape priors. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005. 6

[9] S. Goferman, L. Zelnik-Manor, and A. Tal. Context-aware saliency detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010. 7, 8

[10] L. Gorelick, F. R. Schmidt, and Y. Boykov. Fast trust region for segmentation. In *CVPR*, pages 1714–1721, 2013. 8

[11] P. Kohli, L. Ladicky, and P. H. S. Torr. Robust higher order potentials for enforcing label consistency. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2008. 2, 3, 5

[12] P. Kohli, L. Ladicky, and P. H. S. Torr. Robust Higher Order Potentials for Enforcing Label Consistency. *International Journal of Computer Vision (IJCV)*, 82(3):302324, 2009. 2, 3

[13] P. Kohli and P. H. Torr. Efficiently solving dynamic markov random fields using graph cuts. In *ICCV*, October 2005. 6

[14] V. Lempitsky, A. Blake, and C. Rother. Image segmentation by branch-and-mincut. In *ECCV*, 2008. 1, 2, 6

[15] T. Liu, J. Sun, N.-N. Zheng, X. Tang, and H.-Y. Shum. Learning to detect a salient object. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007. 7

[16] L. Lovász. Submodular functions and convexity. *Mathematical programming: the state of the art*, pages 235–257, 1983. 2

[17] F. Perazzi, P. Krähenbühl, Y. Pritch, and A. Hornung. Saliency filters: Contrast based filtering for salient reg. detect. In *CVPR*, 2012. 7, 8

[18] T. Pock, D. Cremers, H. Bischof, and A. Chambolle. Global solutions of variational models with convex regularization. *SIAM Journal on Imaging Sciences*, 3:1122–1145, 2010. 1

[19] C. Rother, V. Kolmogorov, and A. Blake. Grabcut - interactive foreground extraction using iterated graph cuts. In *ACM transactions on Graphics (SIGGRAPH)*, August 2004. 1, 2, 3, 4, 5

[20] M. Unger, T. Pock, D. Cremers, and H. Bischof. Tvseg - interactive total variation based image segmentation. In *British Machine Vision Conference (BMVC)*, Leeds, UK, September 2008. 1

[21] O. Veksler. Star shape prior for graph-cut image segmentation. In *European Conference on Computer Vision (ECCV)*, 2008. 6

[22] S. Vicente, V. Kolmogorov, and C. Rother. Joint optimization of segmentation and appearance models. In *IEEE International Conference on Computer Vision (ICCV)*, 2009. 1, 2, 4, 5

[23] N. Vu and B. Manjunath. Shape prior segmentation of multiple objects with graph cuts. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008. 6

[24] Y. Zhai and M. Shah. Visual attention detection in video sequences using spatiotemporal cues. In *ACM international conference on Multimedia (ACM Multimedia)*, 2006. 7, 8

[25] S. C. Zhu and A. Yuille. Region competition: Unifying snakes, region growing, and Bayes/MDL for multiband image segmentation. *IEEE transactions on PAMI*, 18(9):884–900, September 1996. 1