Adaptive Figure-Ground Classification

Yisong Chen Antoni B. Chan

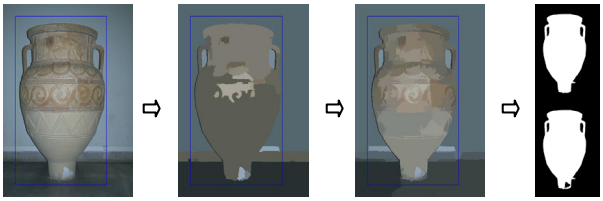
Abstract

This paper proposes a foreground-extraction method from color image by statistically labeling the mosaics created by mean-shift. Under the assumption that a flexibly assigned mask region can provide a good statistical description for the background, we employ the multivariate normal distribution to model the over-segmented patches adaptively generated by mean-shift, estimate the statistical parameters and compute the background priors. The unlabelled patches are sorted and labeled gradually via statistical similarity computation. We propose two probability distances to do similarity measure in a 5D joint color-spatial feature space. Multiple hypotheses under different evaluation criteria are adopted to increase the chance of success. Accurate foreground extraction is achieved with low computational cost. Experiments show that our probabilistic formulation has obvious advantage for multi-connectivity, multi-hole foreground extraction.

1. Introduction

Foreground extraction in still images plays a key role in vision applications [1]. Popular approaches include interactive graph cut [2], random walk [3], geodesic [4], information theory [5], and variational solutions [6].

The biggest problem we are facing might be how to effectively supervise the segmentation and make the routine as intelligent as possible. On the one hand, we are looking for better ways of providing a priori knowledge to guide segmentation by user interaction. Bounding box assigning [7] and seed positioning [8] are two representative interactive methods. On the other hand, we always desire simple models that free users from troublesome algorithm design. Current approaches suffer mainly from the uncertainty of model selection, feature organization [9, 10], parameter tuning [11] and goodness evaluation. Different models lead to different results and there exists no dominant winner [12]. Recent attempts like additional learning process and multiple hypotheses have reported encouraging results [13], although widely applicable solution in the absence of a priori knowledge remains a big challenge.



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

Figure 1. Adaptive figure-ground classification solving pipeline (a) original image & mask; (b) initial ms patches(h_s=7,h_r=6.0) (c) adaptive ms patches(h_s=15,h_r=2.7); (d) D_M and D_K selections

In this work, we present a weakly supervised foreground extraction framework that gives promising solutions to the above 3 questions in a broadly applicable environment. The pipeline of our framework is illustrated in Figure 1. Under the assumption that a carefully assigned mask region is able to provide sufficient statistical information about the background, we treat the task as a figure-ground (f-g) classification on the over-segmented patches generated by the mean-shift algorithm. We model all the region patches as multivariate normal distributions in a 5D joint color-spatial feature space. Two novel probability distances are defined to measure the similarities and new labels are assigned gradually by comparison with known priors. Multiple hypotheses are output to add the chance of success. This scheme avoids the trouble of parameter tuning and makes it possible to fully enjoy the favorable characteristics of the mean-shift algorithm in a direct and intuitive manner. It overcomes many drawbacks of state-of-the-art techniques and generates surprisingly good results for challenging images. *The main contribution is a very simple model equipped with two powerful distance measures, which leads to efficient solving procedure and excellent results*.

2. A foreground-background classification framework

We call our algorithm a weakly supervised one because it merely relies on interactive mask assigning and need no other a prior knowledge. The advantage of a tight bounding box in the context of foreground extraction is obvious as addressed in [14]. So we also take such a bounding box to help define the background priors. We further extend this

concept to treat different foreground and background positions. Briefly speaking, a mask bounding-box is interactively assigned by the user. Then either side of the box can be defined as the background mask. The complement of the background mask makes the foreground mask. This mask definition flexibly handles different cases of partially-inside foreground. This operation is illustrated in Figure 2.

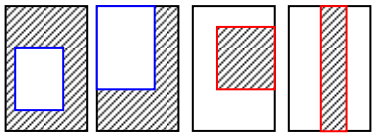


Figure 2. Bounding-box based mask definition.
The background mask (striped region) is defined as the outside of the blue boxes (a) or the inside of the red boxes (b).
The foreground mask (blank region) is defined as the inside of the blue boxes (a) or the outside of the red boxes (b).

2.1. Patch making: mean-shift segmentation

Defining the segmentation as the grouping of non-overlapping regions instead of pixels has become a popular approach due to its advantages in information transfer and computational efficiency [15]. To perform a pre-segmentation there are a lot of candidate algorithms, from the conventional watershed transform to modern over-segmentation schemes like mean-shift [16], GBIS [17] and normalized cut (NCut) [18]. We choose mean-shift as our super-pixel generator because mean-shift patches are easier to describe statistically in comparison to GBIS or NCuts [19]. Moreover, mean-shift is known as an edge-preserving smoothing filter that makes the over-segmentation with detailed boundaries. This makes it possible to focus on the color-spatial features in later processing. In addition, this technique for finding clusters does not require all the points in a cluster to lie within any fixed distance. This is a very useful property that benefits our multi-connectivity, multi-hole oriented segmentation.

The output of the mean-shift preprocessing is a partition of the original image, I, into a set of region patches $R=\{p_1, p_2, \ldots, p_n\}$

$$\bigcup p_i = I, p_i \cap p_j = \Phi \quad (1)$$

Our objective is to group these patches into a foreground category F and a background category B. That is, for every patch p_i we perform a binary classification so that

$$L(p_i) = \begin{cases} 1 & if\ p_i \in F \\ 0 & if\ p_i \in B \end{cases} \quad (2)$$

We directly adopt the mean-shift 5D space as our feature space for similarity measure. In other words, we treat the 3D color features and the 2D spatial features identically and do not give any priority to spatially adjacent patches. We take such a joint organization because we expect that, the feature modes close in position in some dimension within a low-dimensional space may become more sparsely distributed and easier to separate by other dimensions in a high-dimensional space.

Under this formulation, a feature vector in the 5D feature space is given by

$$f = (L, a, b, x, y) \quad (3)$$

where (x y) are the 2D pixel coordinates and (L a b) are the pixel values in the Lab color space. We use the Lab color space because Lab is in general better modeled by normal distribution in comparison to RGB [20]. Mean-shift is a hill-climbing algorithm capable of finding cluster modes in this joint color-spatial feature space by kernel density estimate [21]. Therefore, we model every mean-shift patch as a multivariate normal distribution in the 5D feature space. That is to say, each patch p_i is treated as a Gaussian distribution $N(\mu_i, \Sigma_i)$. The 5D mean vector μ_i and the 5*5 covariance matrix Σ_i are estimated using patch statistics. Before the estimate all the patches are eroded with a radius-1 disk structuring element to avoid border effects.

The result of the mean-shift algorithm relies heavily on the two bandwidth parameters, h_s and h_r. Different initial settings may lead to totally different super-pixel sets and only some of them are suitable for the subsequent classification routine [22]. For instance, for the example image in Figure 1, the default setting $h_s=7$, $h_r=6$ fails to identify the weak edge on the lower right part of the jug and causes the foreground region to "leak" into the background. A desired over-segmentation should generate a reasonable patch set that prevents such leaking effect between foreground and background patches. A good patch set generally corresponds to some appropriate mean-shift bandwidth parameters, which are not known at the beginning of the algorithm but have to be determined by some intelligent module.

Fortunately, the multivariate normal distribution assumption allows us to adaptively guess the bandwidth parameters from some initial distribution statistics. The idea originates from the following theorem proved in [23]:

Theorem: Assume the true distribution of a mean-shift patch is $N(\mu_i, \Sigma_i)$ and the fixed-bandwidth mean shift is computed with a normal kernel K_H. The bandwidth normalized norm of the mean shift vector is maximized when the analysis bandwidth H is equal to Σ.

This theorem characterizes the relationship between the bandwidth parameters and the covariance matrix of the multivariate distribution. This makes it possible to coarsely guess one of them if the other is known. Starting from this idea, we put forward the following method to initialize the mean-shift bandwidth parameters h_s and h_r.

With the default bandwidths $h_s=7$ and $h_r=6$ we do an initial mean-shift segmentation. Then we collect all patches overlapped with the foreground mask region and compute their covariance matrices by patch statistics. The 5*5

covariance matrix Σ of a patch has the form of equation (4).

$$\Sigma = \begin{bmatrix} \Sigma_{rr} & \Sigma_{rs} \\ \Sigma_{sr} & \Sigma_{ss} \end{bmatrix} \quad (4)$$

The upper-left 3*3 submatrix Σ_{rr} corresponds to the covariance matrix in the (L,a,b) subspace, and the lower-right 2*2 submatrix Σ_{ss} corresponds to the covariance matrix in the (x,y) subspace. Then, we employ the following equation to initialize h_s and h_r.

$$h_s = \left\lfloor \sqrt{mean(trace(\Sigma_{ss})/2)} \right\rfloor \; h_r = \sqrt{mean(\max(diag(\Sigma_{rr})))} \quad (5)$$

In brief, under the multivariate Gaussian model h_s and h_r are estimated from the mean square root of the corresponding variance values. The slight difference between h_s and h_r is due to the observation that the variances of the three color components are not of equal importance in the Lab space. The biggest one of the three is in general more dominant.

Although equation (5) is only a coarse estimate it can give a reliable initialization of h_s and h_r in the context of foreground extraction. This estimate can be done iteratively for better performance. However, our experiments show that the iteration is not necessarily convergent and one trial is generally good enough to give a reasonable initialization.

2.2. Similarity measure: a statistical computation

Our framework is established on the assumption that a pre-assigned mask provides sufficient background statistics. In light of this assumption, we do mean-shift again with the adaptive bandwidths, label the patches overlapping with the background mask region as the background priors, and obtain an initial foreground map. The final foreground region is obtained by gradually refining the initial foreground through statistical similarity comparison.

a) original image (b) mean-shift patches (c) correct result

Figure 3. An example foreground object with multiple holes. Patch *A* is a local sample of the global patch *B*. The two patches should be grouped together even if they are not spatially adjacent.

Bhattacharyya distance can be used to quantify the proximity between two statistical samples by estimating the amount of overlap in terms of mean and covariance comparison [24]. In our binary classification context it is not a good choice. Figure 3 illustrates a simple but common example, where a densely distributed patch *A* (the hole) is a local sample of a sparsely distributed patch *B*. Perceptually *A* and *B* should be grouped together even if they are not adjacent. However, the Bhattacharyya distance may be large due to big covariance difference. Therefore, we need some other similarity measures treating such cases well.

The multivariate Gaussian model makes it very easy to measure the probability distance between two mean-shift patches. It is well known that there exists a closed-form Kullback-Leibler(KL) divergence between two Gaussians $N(\mu_1, \Sigma_1)$ and $N(\mu_2, \Sigma_2)$ [25].

$$KL(N_1, N_2) = \frac{1}{2}(\log(|\Sigma_2|/|\Sigma_1|) + Tr(\Sigma_2^{-1}\Sigma_1) + (\mu_1 - \mu_2)^T \Sigma_2^{-1}(\mu_1 - \mu_2)) \quad (6)$$

Equation (6) is not symmetric and thus inconvenient in similarity comparison. To overcome this drawback we suggest the following minimum KL-divergence to measure the statistical distance.

$$D_K(N_1, N_2) = \min(KL(N_1, N_2), KL(N_2, N_1)) \quad (7)$$

Equation (7) is a symmetrized variation of the KL divergence between two Gaussians. It has an intuitive interpretation that the two patches should be grouped together if either of them can be well described by the other.

The computation of the logarithm term of Equation (6) is sometime numerically instable due to unreliable covariance matrixs Σ_1 or Σ_2 caused by singular patches. To remove such instability we also define a more conservative minimum Mahalanobis distance.

$$D_M(N_1, N_2) = \min((\mu_1 - \mu_2)^T \Sigma_2^{-1}(\mu_1 - \mu_2), (\mu_2 - \mu_1)^T \Sigma_1^{-1}(\mu_2 - \mu_1)) \quad (8)$$

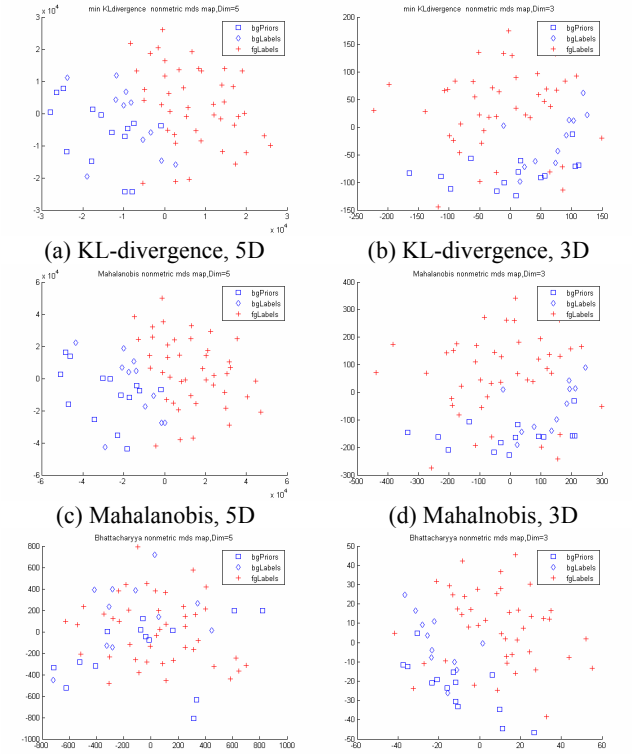
Equation (8) can be deemed as a variation of the minimum KL divergence by retaining only the dominant mode comparison term. Both D_M and D_K treat the mutual "belong to" relationship well and the background holes demonstrated in Figure 3 can be reliably identified. Roughly speaking, there is no guarantee one of them is better than the other. But they indeed provide beneficial complements to each other. Therefore, in our framework we take both similarity measures and output multiple hypotheses.

Provided a similarity metric D (either D_M or D_K), we can define the distance from a single patch *p* to a region set *R* as equation (9), and the distance between two region sets *R1* and *R2* as equation (10).

$$D(p, R) = \min_{r \in R}(D(p, r)) \quad (9)$$

$$D(R_1, R_2) = \min_{r \in R_1}(D(r, R_2)) \quad (10)$$

In Figure 4, we use non-metric multidimensional scaling (MDS) to compare patch dissimilarities under several different feature and distance configurations. We can see that the sample scatter of the KL divergence and the Mahalanobis distance are similar. Moreover, the foreground and the background patches are better separated with the 5D D_M and D_K similarity measures. The figure also reveals that under D_M and D_K the background priors indeed form a statistical representation of the background. This gives a visual validation for the plausibility of our formulation.



(a) KL-divergence, 5D (b) KL-divergence, 3D

(c) Mahalanobis, 5D (d) Mahalnobis, 3D

(e) Bhattacharyya, 5D (f) Bhattacharyya, 3D

Figure 4. MDS results of six different feature and distance configurations for the row-2 image of Figure 7 (68 patches).

2.3. Binary classification: gradual labeling

With the probability distances defined in Section 2.2 we build our figure-ground classification algorithm. For simplicity all discussions in this subsection are based on D_M. The D_K based framework can be similarly established.

A full classification trial is composed of two steps. First, we label the patches sufficiently far from the background priors as foreground patches. Second, we gradually merge the unlabeled patches into the foreground or the background group by comparing their distances to the existing foreground patches and background priors.

For the first step, we use a threshold to determine whether a patch statistic is sufficiently far from the background priors. Namely, we set a patch p as foreground if its distance to the background priors B is greater than a threshold D_t. This is formulated by equation (11).

$$L(p) = 1, if\ D(p,B) > D_t \tag{11}$$

Equation (8) has an intuitive interpretation that two patches are judged as sufficiently far if and only if none of their center modes is within a confidence interval of the other. Equation (11) extends this concept to say that a patch is unlikely to be a background mosaic if it is sufficiently far from all known background distributions.

After the first step we already have a foreground patch group F and a background priors group B, together with the well defined inter-patch similarity matrix. Therefore an energy-minimization framework, such as graph-cut, can be easily organized to accomplish the second step [26]. However, graph constructing and parameter tuning is not easy and brings much uncertainty to the result. Here we take a more straightforward strategy. The unlabeled patches are first sorted in descending order by their distances from the background priors. Then, they are handled in turn by comparing their similarities to the background and the foreground. A label is assigned to the patch according to the comparison result. This is described by equation (12).

$$L(p_i) = \begin{cases} 1 & if\ D(p_i,F) \le D(p_i,B) \\ 0 & if\ D(p_i,F) > D(p_i,B) \end{cases} \tag{12}$$

During the labeling procedure, the foreground group F keeps updating on-the-fly whereas the background group B remains fixed to avoid undesired error propagation. This is again based on the assumption that the original B is already a good enough representation of the background statistics.

The above routine relies on a predefined threshold D_t. for our statistical measure defined in a 5D feature space there often exist more than one appropriate threshold. This leaves much freedom to us to find them.

Multi-segmentation aided parameter tuning is adopted by recent studies and promising results are reported [22]. We take a similar approach and suggest the following method to find a good threshold D_t. Based on the fact that the statistical measure of the closeness from a sample of the multivariate normal population to the center mode is subject to a chi-square distribution [27], we can convert a confidence interval of this chi-square distribution, say, 50.0%~99.9%, to the corresponding probability distance interval $[D_l, D_u]$. This gives a lower bound and an upper bound for the threshold D_t. Then we exhaustively try the interval $[D_l, D_u]$, compute an evaluation score from every segmentation result, and output the most promising solutions. Since our model is a discrete one and the threshold D_t is the only parameter to tune, this 1D brute-force search is within the tractable range and the computational cost is in fact very low.

One final question still remains in the above scheme: How to define the evaluation score that judges the goodness of a solution? Taking into account the fact that the perceptually meaningful segmentation may correspond to different cost functions in different cases, we adopt multi-hypotheses to determine the final output with no priori knowledge needed. Currently our evaluation criteria set contains three score functions that can be easily computed from a candidate solution. They are, respectively, sum-cut, average-cut and maxmin-cut (abbreviated as s-cut, a-cut, and m-cut). Other score functions, like the ones based on edge/shape cues or semantic priors, can be feasibly incorporated to enclose any available prior knowledge [13].

The sum-cut score function is defined as the sum of $D(f,B)$ for all foreground patches f. In other words, we select the threshold D_t that maximizes the following equation.

$$\arg\max_{D_t \in [D_l, D_u]} \sum_{f \in F(D_t)} D(f, B(D_t)) \quad (13)$$

where $F(D_t)$ and $B(D_t)$ are respectively the foreground and the background groups computed from the threshold D_t.

By replacing the sum value with the average value in (13), we get the average-cut score function (14):

$$\arg\max_{D_t \in [D_l, D_u]} \frac{1}{|F(D_t)|} \sum_{f \in F(D_t)} D(f, B(D_t)) \quad (14)$$

Similarly we define the maxmin-cut score function by

$$\arg\max_{D_t \in [D_l, D_u]} D(F(D_t), B(D_t)) \quad (15)$$

The combinatorial property of the formulation determines that the curves of the score functions (13-15) within the interval $[D_l,D_u]$ are not smooth, but piecewise constant. Put it another way, the optimal value generally occurs in a full interval instead of a single point. This fact eases the exhaustive search to a discrete one and we need merely check some key points within the interval $[D_l,D_u]$. We propose the following D_t-solving algorithm. First we sort all values of $D(p,B)$ in ascending order.

$$D_l = d_0 < d_1 < d_2 < \ldots < d_n < d_{n+1} = D_u \quad (16)$$

where n is the number of distances within $[D_l,D_u]$ in $D(p,B)$ and d_i is the i-th smallest value. Then we define a D_t-testing set S_{D_t} that contains all D_t to be tested by

$$S_{D_t} = \cup_{0..n} \{(d_i + d_{i+1})/2\} \quad (17)$$

This method greatly reduces the computational cost in comparison to the exhaustive search.

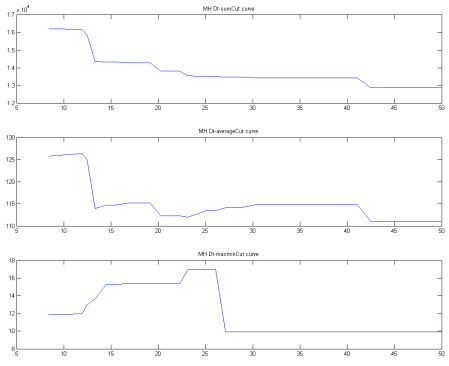


Figure 5. Three D_t-*score* curves for the row-4 image of Figure 7.

We bound the threshold interval loosely with D_l=*5.0*, D_u=*50.0*, which correspond to *0.5841* and *1-10*$^{-9}$ critical values of the 5-dof chi-square distribution. The three score functions are illustrated by the D_t-*score* curves in Figure 5. Roughly speaking, s-cut is a decreasing function and gives a conservative estimate of the solution near D_l, whereas a-cut and m-cut tend to choose larger and more reasonable D_t. Particularly, as addressed in [28] the solutions to m-cut may not be unique (Two m-cut solutions for Figure 5, note the two steps of the s-cut curve within the optimal m-cut interval). Fortunately, Different m-cut solutions in general have similar appearance. Therefore, if this happens, we output only the two m-cut solutions at the left and the right terminals of the solution interval.

3. Experiments

We carry out experiments on three popular datasets to test our segmentation method and report both objective and subjective evaluations. The platform is Intel(R) core(TM) i5 CPU, 2 core processor, 2.8GHz with 8GB RAM, running windows 7 64bit operating system.

3.1. The Weizmann dataset

For the Weizmann evaluation dataset with ground truth segmentations, the results are reported by F-measure criterion, $F = 2PR/(P+R)$, where P and R are the precision and recall values [29]. The algorithm outputs all the selections made by the score functions (13-15) under both D_M and D_K similarity measures, and leaves the final decision to the user. The initial bandwidth parameters are set as h_s=*7*,h_r=*6*. An example output is illustrated in Figure 6, where the D_K a-cut and 2nd m-cut give the best results among eight candidates and are selected by the user.

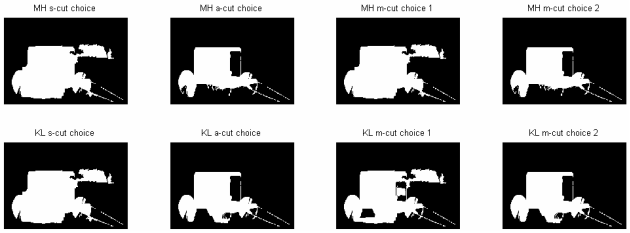
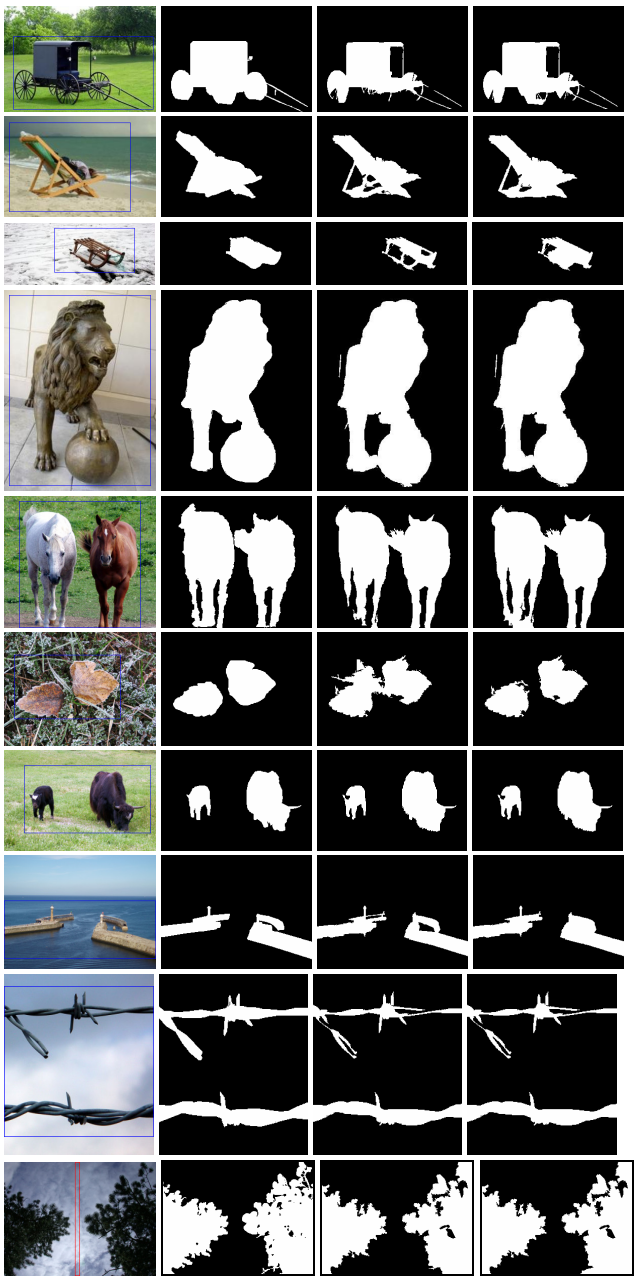


Figure 6. An example output for the row-1 image or Figure 7

The results in Figure 7 and Table 1 discover that the adaptive f-g classification can fit different scenes well. Although the adaptive bandwidths may be significantly different from the initial values, the output is competitive to the best manual initialization. For the 100 1-object images, the best performances reached by D_M and D_K are respectively 76 and 72. The corresponding values for the 100 2-object images are, respectively, 83 and 79. As a whole D_M and D_K are equally good and act as beneficial complements. For both D_M and D_K, all the three score functions defined in Equations (13-15) report surprisingly good results. The selections of D_M and D_K are mostly similar but not always the same. Either of them can be slightly better than the other with almost equal chance. In the absence of a priori knowledge, we suggest outputting all candidates and letting the user make the final decision. The user decision is simulated by $F=max(F_M,F_K)$ in Table 1.



original ground truth D_M selection D_K selection

Figure 7. Weizmann test examples. Rows 1-4 are 1-obj examples. Rows 5-9 are 2-obj examples. D_M performs better for rows 2,3,8; D_K performs better for rows 1,5,6; they are equally good for rows 4,7,9,10. All masks are plotted as blue or red boxes in the original image. Row 10 is an example of the unusual type-b mask.

Equipped with two complementary distance measures D_M and D_K, the adaptive f-g classification is very powerful in labeling background holes or multiple connected components. It even identifies many details missed in the manual-made truths (rows 1, 2, 3, 8, 9 of Figure 7). Therefore, the actual F-measures should be slightly higher than the data in the table.

Table 1. F-measures on the Weizmann and grabcut images [30,31]. $F=max(F_s,F_a,F_m)$ for individuals.

		Weizmann 1-obj	Weizmann 2-obj	Grabcut images
$\overline{F}_s$ (votes)	D_M	0.90 ± 0.015	0.84 ± 0.027	0.88 ± 0.029
	D_K	0.90 ± 0.014	0.83 ± 0.029	0.86 ± 0.032
$\overline{F}_a$ (votes)	D_M	0.88 ± 0.032	0.87 ± 0.029	0.88 ± 0.048
	D_K	0.88 ± 0.037	0.85 ± 0.033	0.87 ± 0.060
$\overline{F}_m$ (votes)	D_M	0.88 ± 0.036	0.88 ± 0.028	0.90 ± 0.048
	D_K	0.88 ± 0.040	0.88 ± 0.029	0.88 ± 0.061
$\overline{F}$ (votes)	D_M	$0.93 \pm 0.010(76)$	$0.89 \pm 0.021(83)$	$0.93 \pm 0.017(33)$
	D_K	$0.93 \pm 0.010(72)$	$0.89 \pm 0.021(79)$	$0.94 \pm 0.017(34)$
$\max(F_M, F_K)$		0.93 ± 0.010	0.90 ± 0.021	0.94 ± 0.016

3.2. The grabcut dataset

For the 50 grabcut test images, we compare our method to the grabcut algorithm under the same bounding box settings. For the adaptive f-g classification the mean-shift bandwidth parameters are initialized by $h_s=7$, $h_r=6$. The F-measures are reported in the last column of Table 1. The grabcut vs. f-g classification scatter plots for the 50 individual F-measures are given in Figure 8. The results show that f-g classification slightly outperforms grabcut.

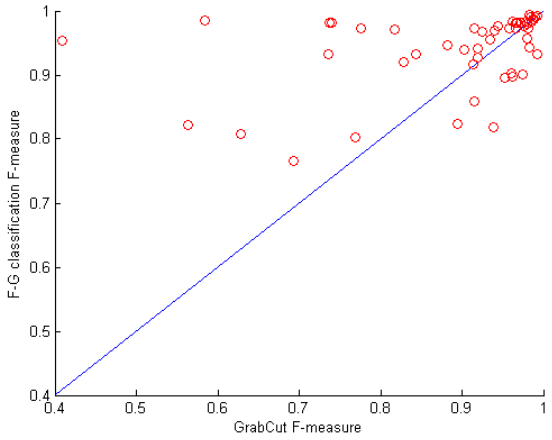


Figure 8. The scatter plot for individual F-measure comparison

Some images in the grabcut image set have very cluttered background but relatively simple foreground. For these images it is not easy to fully describe the background by a single bounding box. We solve this problem by switching the roles of foreground and background. Namely, at the initialization stage we take the foreground region as the background and assign a bounding box fully enclosed by the foreground region. After the segmentation we reverse the foreground and the background and obtain the final result. This operation can also be adopted by the grabcut algorithm. Figure 9 gives an example image that can be well treated by the switch operation. This example gives a rule of thumb for mask selection. That is, the background mask should be statistically simple and easily characterized by a bounding box.

(a)Original image (b) f-g classification result (c) grabcut result
Figure 9. A figure-ground switching example.
The green box is a type-c mask indicating figure-ground switch.

Table 2 compares the execution time and the mean F-measures between the grabcut algorithm and the adaptive f-g classification for all the three image sets of section 3.1 and 3.2. Clearly the adaptive f-g classification performs better in both segmentation quality and executing speed.

Table 2. Performance comparison of grabcut and f-g classification

		Weizmann 1-obj	Weizmann 2-obj	Grabcut dataset
time	grabcut	6.53 s	4.86 s	16.53 s
	f-g classification	**3.54 s**	**2.54 s**	**11.56 s**
$\bar{F}$	grabcut[7]	0.85 ± 0.035	0.80 ± 0.046	0.89 ± 0.036
	f-g classification	$\mathbf{0.93 \pm 0.010}$	$\mathbf{0.90 \pm 0.021}$	$\mathbf{0.94 \pm 0.016}$

3.3. The Berkeley segmentation dataset

In this part we evaluate our method on the Berkeley segmentation dataset [32]. Figure 10 gives the segmentation results of some challenging images in the Berkeley dataset (rows 1,2) and the grabcut dataset (row 3). The adaptive bandwidth parameters $[h_s,h_r]$ computed by Equation (5) are given for the Berkeley examples. The adaptive initialization works well and reliably treats multi-connectivity, multi-hole scenes. The minimum KL divergence D_K and the minimum Mahalanobis distance D_M make beneficial complements and greatly raise the chance of finding good segmentations.

The experiments reveal that the f-g classification method robustly propagates the boundary priors into the foreground mask region and *reliably treats multi-connectivity, multi-hole scenes*. As a typical example, almost all connected components and all holes in image 370036 are successfully identified. Such scenes are difficult for other schemes unless additional efforts are involved.

3.4. Future work

The adaptive figure-ground classification is a highly automatic foreground extraction framework and is able to reach a reasonable solution in many scenes. However, for some cluttered scenes the automatically estimated bandwidths might not be optimal. This can be solved by adding some texture or shape constraints in feature organization and similarity measure. Moreover, even a good mean-shift initialization may still leave some boundary spikes. This can be remedied by additional morphological or matting operations [33], which is another direction of future work. Finally, currently only color and spatial cues are used in our classification framework. How to expand the framework to effectively contain other features, like edges or textures, for problem formulation or goodness evaluation, remains an open question.

4. Conclusion

An adaptive figure-ground classification algorithm is proposed to automatically do foreground extraction from bounding-box based background priors. The similarity measure is defined as the probability distance between adaptively generated mean-shift patched in a 5D feature space. The background statistics defined by a mask box is conveniently propagated through the region of interest. Multiple hypotheses based on different score functions are employed to add the chance of success. The experiments show that this method is very promising by achieving great success for multi-connectivity, multi-hole scenes.

References

[1] V. Gulshan, C. Rother, A. Criminisi, A. Blake and A. Zisserman, Geodesic star convexity for interactive image segmentation. In CVPR2010, pp. 3129-2136.

[2] Y. Boykov and M. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images. In Proc. ICCV2001, volume 1, pp. 105-112.

[3] L. Grady. Random walks for image segmentation. PAMI, 28(11):1768–1783, 2006.

[4] X. Bai and G. Sapiro, A geodesic framework for fast interactive image and video segmentation and matting. In ICCV2007, pp. 1-8.

[5] S. Bagon, O. Boiman, and M. Irani, What is a good image segment? a unified approach to segment extraction. In ECCV, pages 30–44, 2008.

[6] G. Hua, Z. Liu, Z. Zhang, and Y. Wu, Iterative local global energy minimization for automatic extraction of objects of interest, PAMI, 28(10), 1701-1706.

[7] C. Rother, V. Kolmogorov, and A. Blake, "grabcut": interactive foreground extraction using iterated graph cuts. ACM Trans. Graph., 23(3):309–314, 2004.

[8] B. Micusik and A. Hanbury. Automatic image segmentation by positioning a seed. ECCV2006, Vol. 2, 468–480.

[9] R. B. S. Alpert, M. Galun and A. Brandt. Image segmentation by probabilistic bottom-up aggregation and cue integration. In CVPR2007, pp. 1–8.

[10] B. Price, B. Morse, and S. Cohen, Geodesic Graph Cut for Interactive Image Segmentation. CVPR2010, pp.3161-3168.

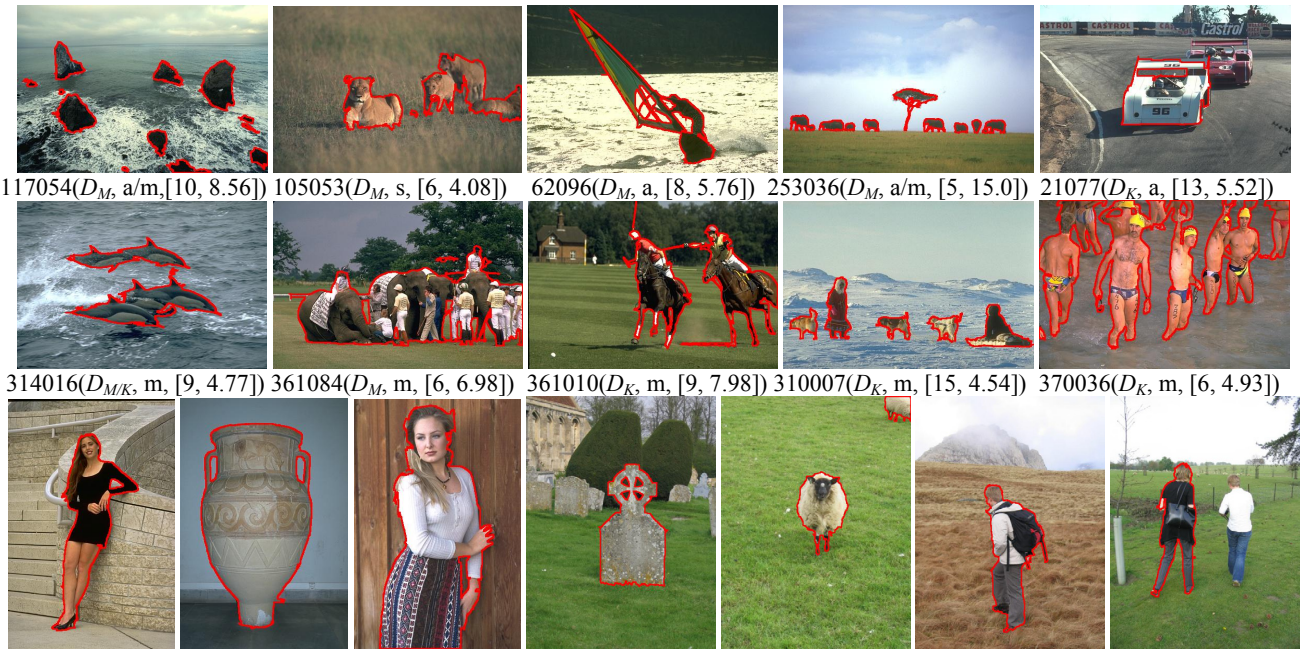


Figure 10. Example Berkeley and grabcut images selected by D_M or D_K (s: voted by s-cut; a: voted by a-cut; m: voted by m-cut). Almost all holes and connected components are successfully identified. D_M and D_K work equally well and make good complement.

[11] A. Blake, C. Rother, M. Brown, P. Perez, and P. Torr. Interactive image segmentation using an adaptive GMMRF model. In ECCV, LNCS 3021, pages 428–441, 2004.

[12] A. Criminisi, T. Sharp, and A. Blake. GeoS: Geodesic image segmentation. In ECCV, pages 99–112, 2008.

[13] J. Carreira and C. Sminchisescu. Constrained parametric min cuts for automatic object segmentation. CVPR2010, pp. 3241-3248.

[14] V. Lempitsky, P. Kohli, C. Rother, and T. Sharp, Image segmentation with a bounding box prior. ICCV2009, pp. 277-284.

[15] Y. Li, J. Sun, C. Tang, H. Shum: Lazy snapping. ACM Trans. Graph., 23(3):303–308, 2004.

[16] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. PAMI, 24(5):603–619, 2002.

[17] P. Felzenszwalb and D. Huttenlocher. Efficient graph-based image segmentation. IJCV, 59(2):167–181, 2004.

[18] J. Shi and J. Malik. Normalized cuts and image segmentation. PAMI, 22(8), pp. 888-905, 2000.

[19] M. Donoser, M. Urschler, M. Hirzer, and H. Bischof. Saliency driven total variation segmentation. In ICCV2009, pp. 817 - 824.

[20] S. R. Rao, H. Mobahi, A. Y. Yang, S. S. Sastry, and Y. Ma. Natural image segmentation with adaptive texture and boundary encoding. In ACCV2009, pp. 135-146.

[21] M. Perpinan, Gaussian mean-shift is an EM algorithm, PAMI, 29(5), 2007, pp. 767-776.

[22] T. Malisiewicz and A. Efros. Improving spatial support for objects via multiple segmentations. In BMVC, 2007.

[23] D. Comaniciu, An algorithm for data-driven bandwidth selection, PAMI, 25(2), 2003, pp. 281-288.

[24] M. Donoser and H. Bischof. ROI-SEG: Unsupervised color segmentation by combining differently focused sub results. CVPR2007, pp. 1-8.

[25] J. Goldberger, S. Gordon, and H. Greenspan, "An efficient image similarity measure based on approximations of KL-divergence between two gaussian mixtures, ICCV 2003, Nice, October 2003, vol. 1, pp. 487–493.

[26] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother. A comparative study of energy minimization methods for markov random fields with smoothness-based priors. PAMI, 30(6):1068–1080, 2008.

[27] K. Kanatani, Statistical Optimization for Geometric Computation: Theory and Practice, Elsevier 1996; reprinted Dover 2005.

[28] A. K. Sinop and L. Grady. A seeded image segmentation framework unifying graph cuts and random walker which yields a new algorithm. In ICCV2007, pp.1-8.

[29] M. Maire, Simultaneous segmentation and figure/ground organization using angular embedding, ECCV2010.

[30] http://www.wisdom.weizmann.ac.il/~vision/Seg_Evaluation_DB/scores.html, Weizmann dataset webpage.

[31] http://research.microsoft.com/en-us/um/cambridge/projects/visionimagevideoediting/segmentation/grabcut.htm, Grabcut dataset webpage.

[32] http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench/,Berkeley segmentation dataset page.

[33] J. Wang and M. Cohen Image and video matting: a survey, Foundation and trends in computer graphics and vision, Vol. 3, No. 2. (2007), pp. 97-175.