Object-level Image Segmentation Using Low Level Cues

Hongyuan Zhu, Jianmin Zheng,
Jianfei Cai, *Senior Member, IEEE* and Nadia M. Thalmann

***Abstract*—This paper considers the problem of automatically segmenting an image into a small number of regions that correspond to objects conveying semantics or high-level structure. While such object-level segmentation usually requires additional high-level knowledge or learning process, we explore what low level cues can produce for this purpose. Our idea is to construct a feature vector for each pixel, which elaborately integrates spectral attributes, color Gaussian Mixture Models and geodesic distance, such that it encodes global color and spatial cues as well as global structure information. Then we formulate the Potts variational model in terms of the feature vectors to provide a variational image segmentation algorithm that is performed in the feature space. We also propose a heuristic approach to automatically select the number of segments. The use of feature attributes enables the Potts model to produce regions that are coherent in color and position, comply with global structures corresponding to objects or parts of objects and meanwhile maintain a smooth and accurate boundary. We demonstrate the effectiveness of our algorithm against the state-of-the-art with the dataset from the famous Berkeley benchmark.**

***Index Terms*—Image segmentation, low level cues, object segmentation, variational model.**

I. Introduction

This paper deals with the process of automatically segmenting an image into a small number of regions. Different from conventional multi-label segmentation that often results in over-segmentation, our target is a small set of regions that have a relatively large size and correspond to objects or parts of objects conveying some semantics or high-level structure/features, in addition to certain homogeneity. The underlying reason for this target is that most images can be viewed as combinations of objects by nature and for an image what attract most of the attention usually are only a few salient objects. Such segmentation is useful in many computer vision tasks such as object recognition and scene understanding, as demonstrated in [3] where images are transformed into the Blobworld representation composed of a small set of image regions and these regions are then used for image retrieval and querying. We hence call our process object-level segmentation. It is similar to semantic segmentation, but we do not consider category-specific labeling that is often involved in the conventional semantic segmentation.

The challenge with such object-level segmentation lies in the fact that though human brain is good at abstracting semantically meaningful regions from visual cues, developing an automatic algorithm that well mimics this brain function is still very difficult. To perceive an image, elements must be perceived as a whole, but most images are currently represented based on local low-level features. In addition, the concepts of "semantics" and human perception are quite subjective and content dependent. In general, automatically generating object-level segmentation is an ill-posed problem. It often requires global image information or high level knowledge, which may come from user input in interactive or semi-supervised methods [4] or labeled database in learning/training based methods [5]. However, we observe that high level knowledge and low level cues are not totally independent and actually some semantics are conveyed in various low level cues. For example, cognition study [6] shows that human vision views part boundaries at those with negative minima of curvature and the part salience depends on three factors: the relative size, the boundary strength and the degree of protrusion. Gestalt theory and other psychological studies have also developed various principles reflecting human perception, which include: (1) human tends to group elements which have similarities in color, shape or other properties; (2) human favors linking contours whenever the elements of the pattern establish an implied direction. Therefore it is interesting to explore what low level cues can produce for high level object segmentation.

Extensive research has been conducted for image segmentation. A broad family of methods makes use of local features such as color and texture for clustering. Examples are MeanShift [7] and Graph-Based Region Merging [2]. While these methods are usually very fast, they tend to produce over-segmentation. Alternatively, methods like spectral clustering [8] and Normalized Cut [1] use eigenvectors for the clustering process. It has been shown that eigenvectors resulting from spectral clustering carry global contour information and thus these methods are able to capture semantic regions with considerable sizes. The gPb-owt-ucm method [9] combines multiple local cues into a globalization process using spectral clustering and then constructs a hierarchical region tree from a set of contours to achieve hierarchical image segmentation, which has demonstrated the state-of-the-art segmentation performance. The segmentation results with these methods are generally good. However, some visual artifacts can still be observed. For example, the region contours do not follow object boundaries very well and the large uniform or smooth regions may be split. On the other hand, the multi-label segmentation can also be formulated as a variational problem [10], [11], [12], [13], [14]. Variational methods have become popular since they can produce smooth and accurate region boundaries

H. Zhu, J. Zheng, J. Cai and N. M. Thalmann are with School of Computer Engineering, Nanyang Technological University, Singapore, email: {hzhu1, asjmzheng, asjfcai, nadiathalmann}@ntu.edu.sg. Contact author: Jianfei Cai.

(a) Original image (b) Ncut (c) Felz-Hutt (d) Potts with RGB (e) Our Method

Fig. 1. (b) & (c): The existing low-level cues based segmentation methods such as Ncut [1] and Felz-Hutt [2] often over-segment the image or cannot align well with true object boundary. (d): The original Potts model relying on local color feature mistakenly treats part of the mountain and chair, the ground and the sky as the same segments. (e): Our method without help of high level knowledge can achieve object-level segmentation to some extent and meanwhile obtain smooth and accurate boundaries. The color in each image just distinguishes regions.

and many fast numerical solvers have been developed. In addition, many of these solvers can be parallelized, which is very suitable for GPU implementation. However, variational methods are in general sensitive to the initializations and bad initialization can result in local minimum [13].

Inspired by the recent progress in image segmentation (especially variational segmentation), we propose to select effective low level cues of images that reflect global color feature, spatial information, and structure information as well and integrate them to form feature vectors. We then substantiate the Potts model with the feature vectors to provide a variational segmentation algorithm. In this way, we can achieve object-level segmentation to some extent without the help of high level knowledge and meanwhile obtain smooth and accurate segmentation boundaries (see Fig. 5). To the best of our knowledge, there was no such work for image segmentation before. The novelties of this paper include:

- Spectral attributes, color Gaussian Mixture Models and geodesic distance from the state-of-the-art techniques are selected as low level cues and elaborately integrated to construct feature vectors for image pixels. These feature vectors encode global color and spatial features as well as global structure information.
- An automatic multi-label segmentation algorithm based on the Potts variational model using the feature vectors is developed, which can produce a small number of regions reflecting local color and spatial coherence and global semantic structure in one framework.
- A heuristic approach is proposed to determine the number of regions based on the stability of *Ncut* values.

II. Related Theory

A. Normalized cut

Normalized cut [1] finds a segmentation that minimizes the so-called *Ncut* value which is defined by the weights of boundary edges between clusters and the weights of all edges within each cluster. The basic idea in normalized cut is that big clusters have large weights within them and minimizing *Ncut* encourages all such weights to be about the same, thus achieving a "balanced" clustering.

Finding the normalized cut is an NP-hard problem. Usually, an approximate solution is sought by finding the eigenvectors of the generalized eigenvalue system $(D-W)v = \lambda Dv$, where $W = [w_{ij}]$ is an affinity matrix of an image graph with w_{ij} describing the pairwise affinity of two pixels and $D = [d_{ij}]$ is a diagonal matrix with $d_{ii} = \sum_j w_{ij}$. Then the segmentation is achieved by recursively bi-partitioning the graph using the first nonzero eigenvalue eigenvector [1] or spectral clustering of a set of eigenvectors [8].

As eigenvectors convey global structure information, normalized cut is less likely to produce small or trivial regions than those methods that just use local statistics. However, a simple clustering of eigenvectors often splits uniform or smooth regions [9]. Our work also uses eigenvectors as spectral attributes for segmentation. We choose an appropriate number of eigenvectors and apply the continuous Potts model to produce segmentation that aligns with global salient edges.

B. The Potts model

The Potts model originates from statistical mechanics [15] and has been widely used in various computer vision tasks such as image de-noising and segmentation [12], [16], [13], [14]. Given an image $I : \Omega \to R$, the Potts model attempts to partition the image into n disjoint sub-regions $\{\Omega_i\}_{i=1}^n$ with $\bigcup_{i=1}^n \Omega_i = \Omega, \Omega_k \bigcap \Omega_l = \emptyset, \forall k \neq l$ by minimizing the functional:

$$\min_{\{\Omega_i\}_{i=1}^n} \sum_{i=1}^{n} \int_{\Omega_i} p(l_i, x)dx + \alpha \sum_{i=1}^{n} C_i(x)|\partial\Omega_i|, \qquad (1)$$

The first term of (1) is the region term that measures the cost to assign label l_i to the data. A simple region term is given by

$$p(l_i, x) = |I(x) - c_i|, i = 1, ..., n \qquad (2)$$

where c_i corresponds to the mean intensity of region with label l_i. The second term of (1) is the boundary term where $|\partial\Omega_i|$ is the perimeter of region Ω_i, and $C_i(x)$ is an edge detector function which is defined by

$$C_i(x) = \frac{1}{1 + |\nabla I(x)|^2}. \qquad (3)$$

The region term and the boundary term are balanced by a tradeoff factor α. Minimizing the region term ensures the segmentation complying with some region coherence and minimizing the boundary term favors the segmentation with tight and smooth boundaries along the salient edges in the image.

By introducing an indicator function $u_i(x)$ for each region $\Omega_i, i = 1...n$,

$$u_i(x) = \begin{cases} 1, & x \in \Omega_i \\ 0, & x \notin \Omega_i \end{cases} \tag{4}$$

the Potts model (1) can be rewritten as

$$\min_{u_i(x)\in\{0,1\}} \sum_{i=1}^{n} \int_{\Omega} \{u_i(x)p(l_i, x) + \alpha C_i(x)|\nabla u_i|\}\, dx, \tag{5}$$

$$s.t. \sum_{i=1}^{n} u_i(x) = 1, \forall x \in \Omega. \tag{6}$$

Whereas this model is nonconvex due to the binary configuration of $u_i(x) \in \{0, 1\}$, currently popular approaches [17][18][19] relax the binary constraint to the interval $[0, 1]$ and approximate (5) with the convex model:

$$\min_{u_i(x)\in[0,1]} \sum_{i=1}^{n} \int_{\Omega} \{u_i(x)p(l_i, x) + \alpha C_i(x)|\nabla u_i|\}\, dx, \tag{7}$$

$$s.t. \sum_{i=1}^{n} u_i(x) = 1, \forall x \in \Omega. \tag{8}$$

1) Multi-label continuous max-flow method: To solve the convex Potts model (7), Yuan *et al.* present a multi-label continuous max-flow formulation [14]. The method maps the functional to n parallel copies $\Omega_i, i = 1...n$, of Ω which are linked to a common source node s and sink node t, where n is the number of labels. Then the minimization problem is transformed to the problem of finding $p_s, p = [p_1, p_2, .., p_n], q = [q_1, q_2, .., q_n], u = [u_1, u_2, .., u_n]$ for $\max\limits_{p_s,p,q} \min\limits_{u} L_c(p_s, p, q, u)$ where

$$\begin{aligned} L_c(p_s, p, q, u) = \int_{\Omega} [p_s + \sum_{i=1}^{n} \langle u_i, divq_i - p_s + p_i\rangle] dx \\ -\frac{c}{2} \sum_{i=1}^{n} \|divq_i - p_s + p_i\|^2 \\ s.t. |q_i(x)| \le \alpha C_i(x), i = 1, ..., n, \forall x \in \Omega \\ p_i(x) \le p(l_i, x), i = 1, ..., n, x \in \Omega \end{aligned}$$

with constant $c > 0$. In this new formulation, for each position $x \in \Omega$, $p_s(x)$ is the flow stream from the source s to x at each copy Ω_i, $p(l_i, x)$ serves as the capacity of the sink flow $p_i(x)$ directed from x at Ω_i to the sink t, $\alpha C_i(x)$ is the capacity of spatial flows $q_i(x)$ defined within each Ω_i and $u_i(x)$ is the indicator function for each label i and works as the Lagrangian multiplier. An algorithm based on the augmented Lagrangian method is introduced in [14] to find the solution. Final segmentation is formed by assigning each pixel x to the label i whose corresponding indicator function $u_i(x)$ has the largest magnitude. It has been shown that compared with previous methods[20][19], the continuous max-flow method has a fast convergence rate and moreover it can be highly parallelized to achieve even faster processing speed. This algorithm is adopted in our approach to solve the Potts model.

III. Proposed method

Given an input image, our goal is to automatically partition the image into a small number of regions that are coherent in color and structure. We divide this problem into two subproblems. The first one is how to segment the image into k regions for a given number k. The second one is how to automatically choose k, the number of regions, which will be described in Section III-D. Combining the solutions to both subproblems leads to an automatical variational image segmentation algorithm.

For the first subproblem, our basic idea is to construct some feature vectors from various low level cues for images and then formulate the Potts model in terms of the feature vectors for segmentation. Basically, this involves the following processes:

- **Global Structure/Boundary Feature Extraction:** Construct eigenvectors and globalized probability of boundary for each pixel which is later used to construct the Potts functional.
- **EM Initialization:** Model the distribution of color and eigenvectors with a k mixture of Gaussians and use the EM method to generate initial means for the Potts model.
- **Functional Formulation and Segmentation:** The global features that consist of global structure, color and spatial information are used to formulate the region and boundary terms of the Potts model to make the segmentation produce homogeneous regions and snap to accurate boundaries. The final segmentation is completed by solving the variational model using the continuous max-flow method [14]

These processes are elaborated in the next three sub-sections.

A. Global Structure and Boundary Feature Extraction

The Potts model works well under the assumption that the image is roughly piecewise constant. However low-level local features such as intensity, color, texture and curvature may not necessarily possess such characteristics, which makes the model sometimes produce trivial regions and false boundaries. Thus it is necessary to consider features that can better describe the underlying data. The eigenvectors resulting from an affinity matrix are such features because they carry global contour information, reflect significant structures and tend to be roughly piecewise constant, which can be observed in Figure 2.

To construct the eigenvectors, we basically adapt the work of [9] that describes a very nice globalization method for contour detection and spectral clustering. The main steps are as follows.

First, a multiscale extension mPb of the posterior probability of boundary at each image pixel x is computed, which considers gradients at different scales for image brightness, color, and texture channels in order to detect both fine and coarse structures. Second, a sparse symmetric affinity matrix $W = [w_{ij}]$ is constructed using the intervening contour cue [21],

[9] and the maximal value of mPb along a line connecting two pixels x_i and x_j with $w_{ij} = \exp(-\max_{p \in \overline{x_i x_j}}\{mPb(p)\}/\rho)$, where $\overline{x_i x_j}$ is the line segment connecting x_i and x_j and ρ is a constant (which is set to $\rho = 0.1$ in literature). Third, let $D = [d_{ij}]$ be a diagonal matrix with $d_{ii} = \sum_j w_{ij}$ and solve for the generalized eigenvectors of $(D - W)v = \lambda Dv$. We choose l eigenvectors corresponding to the l smallest nonzero eigenvalues. l is determined via an eigen-gap heuristic [22]. Finally, a spectral boundary detector sPb is defined at each pixel by the convolutions of each eigenvector with Gaussian directional derivative filters. As pointed out in [9], signal mPb fires at all the edges and spectral signal sPb extracts only the most salient edges in the image. Thus a linear combination of mPb and sPb is suggested to yield a globalized probability of boundary, gPb. In our application, from this process we extract l eigenvectors and the gPb map, which will be used in the subsequent steps.

B. EM Initialization

One drawback of the Potts model is that it is sensitive to the choice of mean $c_i, i = 1, .., n$, in (2). Inappropriate initialization of c_i can make the model get stuck in local minimum [13]. Therefore here we propose a simple and efficient method to generate reasonable initial means.

As eigenvectors are nearly piecewise constant, thus for $k-$partition, we can assume they are drawn from k Gaussians in the mixture model and use the EM algorithm to determine the maximum likelihood parameters of the mixture of k Gaussians in the feature space. For each pixel, we construct a feature vector of length $(l+3)$ that consists of RGB colors and eigenvectors $v(x) = (R, G, B, e_1, \cdots, e_l)$. Then we perform the EM algorithm to estimate the parameters.

To apply the EM algorithm, we need to initialize parameters. The initial mixing weights π_i are simply set to $\frac{1}{k}$. The covariance matrices Σ_i are set to the identity matrix. For the means $\mu_1, \mu_2, ..., \mu_k$, we run K-means to generate k clusters and then use the means of these clusters as the initial means. After the EM iteration stops, each pixel is assigned to the label corresponding to the largest probability, thus delivering k initial regions.

Note that Carson et al. [3] use a similar method to generate blobworld for image retrieval. However the low-level feature (color, Gabor filter and pixel location) can prevent EM from approaching semantic objects. In addition, parameter initialization in their work is done in an ad-hoc manner which can lead to failure detection of the object of interest.

C. Variational Segmentation

The Potts model contains two terms: a region term (2) and a boundary term (3). Below we show how to formulate these two terms to incorporate global information such as the spectral attributes.

1) Region term formulation: In [14], the variational model is defined in the RGB space. When natural images contain much variation of colors, local features such as color and texture are often insufficient to reflect the structure of the images. To generate a good partition, it is desirable to include some global information in the region term.

In our case, the EM clustering has provided a good approximation of k regions, which could be utilized to extract some useful region information for later variational segmentation. We first introduce the GMM to describe the color distribution of each label, which has demonstrated its success in interactive segmentations [4], [23]. Specifically, for each pixel x in the image, we can obtain a set of probabilities $[g_1(x), ..., g_k(x)]$, where $g_i(x)$ denotes the probability that the pixel belongs to label l_i and is computed by

$$g_i(x) = \frac{-\log Pr(x|\bar{l_i})}{-\log Pr(x|l_i) - \log Pr(x|\bar{l_i})} \tag{9}$$

where $Pr(x|l_i)$ indicates the probability that pixel x fits the GMM of label l_i and $Pr(x|\bar{l_i})$ is the probability that x fits the GMM of any label other than l_i.

To strengthen the region information when foreground and background colors are not well separable, we further introduce the geodesic probability to describe the spatial information of the seed regions. The geodesic probability indicates the likelihood of pixel x belonging to label l_i and is defined by [24]

$$\epsilon_i(x) = \frac{D(x, \bar{l_i})}{D(x, l_i) + D(x, \bar{l_i})} \tag{10}$$

where $D(x, l_i)$ is the geodesic distance from pixel x to the seed region of label l_i and $D(x, \bar{l_i})$ is the geodesic distance from x to other seed regions.

Now we are ready to construct a feature vector for each pixel x. The feature vector is a $(2k + l)$ vector:

$$[g_1(x), ..., g_k(x), \epsilon_1(x), ..., \epsilon_k(x), e_1(x), ..., e_l(x)] \tag{11}$$

where k is the number of labels, l is the number of the selected eigenvectors generated in Section III-A, $g_i(x)$ is the GMM probability of label i defined by (9), $\epsilon_i(x)$ is the geodesic probability of label i defined by (10), and $e_i(x)$ is the element corresponding to x in the $i-$th eigenvector. The feature vector can be viewed as a point in a $(2k + l)-$dimensional space called the feature space. The Potts model will be applied to this space. Thus we define the region function for each label l_i to be

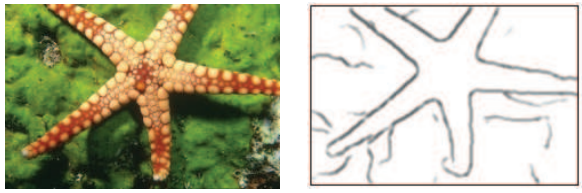
$$\begin{aligned} p(l_i, x) = \alpha_1^i \cdot [\alpha_2^i \cdot (1 - g_i(x)) + (1 - \alpha_2^i) \cdot \\ (1 - \epsilon_i(x))] + (1 - \alpha_1^i) \cdot d_{eigen}(l_i, x) \end{aligned} \tag{12}$$

where α_1^i and α_2^i are the tradeoff factors for label l_i,

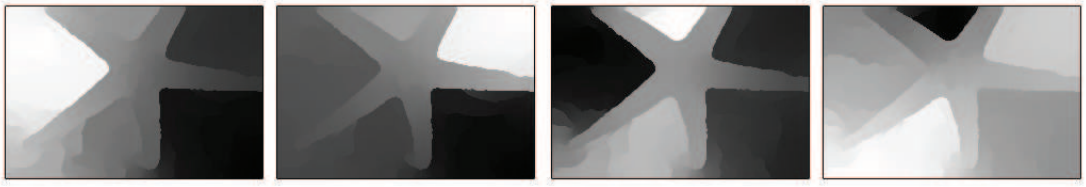
$$d_{eigen}(l_i, x) = \sqrt{(e_1(x) - e_1^{l_i})^2 + \cdots + (e_l(x) - e_l^{l_i})^2}$$

and $e_j^{l_i}$ is the mean of spectral attribute e_j for label l_i. The region function (12) has three terms: the first two describe the color and spatial information of regions and the third one describes the global structure information. The combination of them enhances the capability of differentiating regions.

It is important to properly set the tradeoff factors α_1^i and α_2^i. When GMMs provide enough information to distinguish one label from the others, the first term should dominate; Otherwise, eigenvectors and the geodesic probability should



(a) Original image (b) gPb Map



(c) Eigenvectors

Fig. 2. (a), (c): An image and its top four non-zero eigenvalue eigenvectors. (b): The gPb contour map generated from eigenvector captures clean and salient image boundaries

play a major role. Thus, as suggested in [4], we set α_1^i and α_2^i to be the Kullback-Leibler divergence between the current label's GMM and the rest labels' GMMs:

$$\alpha_1^i = \alpha_2^i = \frac{1}{n}\sum_{j=1}^{n}\left|\frac{\log Pr(x_j|l_i) - \log Pr(x_j|\bar{l_i})}{\log Pr(x_j|l_i) + \log Pr(x_j|\bar{l_i})}\right| \quad (13)$$

where $\bar{l_i}$ indicates the rest labels and n is the number of pixels.

An example which compares different combinations of global features is shown in Figure 3. We can see that by using GMM, geodesic probability and eigenvectors, the result is more meaningful and accurate than using the local RGB information.

2) Boundary term formulation: The boundary term in (1) is a weighted total variation of function u. The weight $C_i(x)$ plays an important role. The definition of $C_i(x)$ in (3) favors the segmentation along the curves where the edge detection function takes small values. In our algorithm, we use the gPb proposed in [25], [9] as the base map where the edge detector of (3) is applied. gPb is computed in the process of generating spectral attributes and it has proved to be powerful signal for edge information. Unlike other classical detectors, gPb makes use of the global information encoded in eigenvectors and thus it can capture the salient edges. However, gPb has limitations in that some weak edges may be missed due to the fact that eigenvectors may not capture small structures. Thus we propose to further incorporate the GMM probability map to enhance the edge detection:

$$C_i(x) = \beta^i \cdot g_c^i(x) + (1-\beta^i)\cdot g_e(x) \quad (14)$$

where g_c^i and g_e are the results of applying the edge detector of (3) to the GMM probability map g_i of (9) and the gPb map respectively, and β^i is a tradeoff factor which is defined in a similar way as α_1^i or α_2^i given in (13).

Figure 4 compares the results of our method with and without g_c^i. We can find that by incorporating the GMM probability maps, the weak edges are enhanced and the segmentation snaps to the salient image boundaries better. In addition, the result with the canny edge detector is the worst, as it captures too much trivial edges which make the algorithm snap to unsalient ones.

D. Selecting the number of regions

We now discuss how to choose k, the number of regions. An ideal value of k should best fit the number of groups present in the image. However, the notion of best fitting is quite subjective. Here we present a heuristic approach to compute k based on the stability of the Ncut values.

Our observation is that if a good k-partition has been formed, increasing the number of segments to $k+1$ will cause the existing segments to be split and merged to form a new segmentation, which usually results in a big change of the Ncut values. This suggests a brute-force approach: perform clustering and compare the Ncut values to select among different values of k. Considering that our goal here is to find the number of regions, we just use the EM clustering, based on which we perform Ncut value stability analysis. Particularly, we choose the best k to be

$$\underset{i\in[2,15]}{arg\,max}\ \{|Ncut(i+1) - 2Ncut(i) + Ncut(i-1)|\} \quad (15)$$

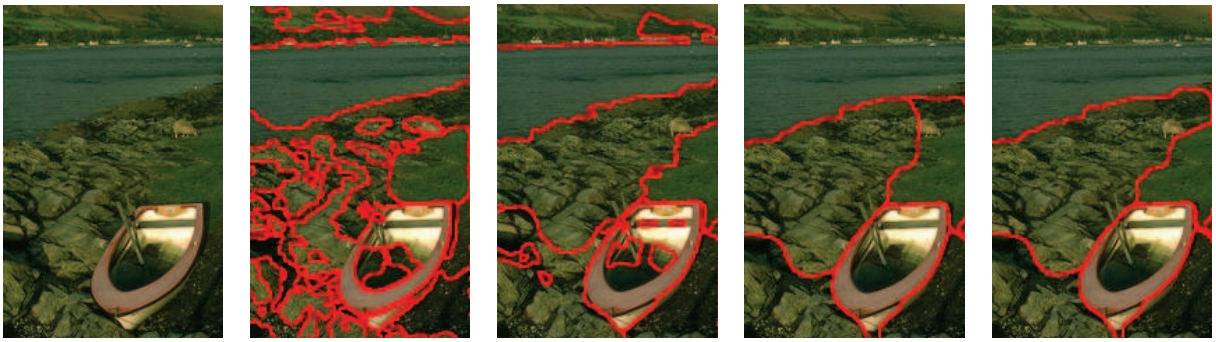
which maximizes the second order difference.

Experimental results shown in Figure 5 demonstrate that the number of regions determined by this heuristic approach leads to meaningful segmentations.

IV. Experiments

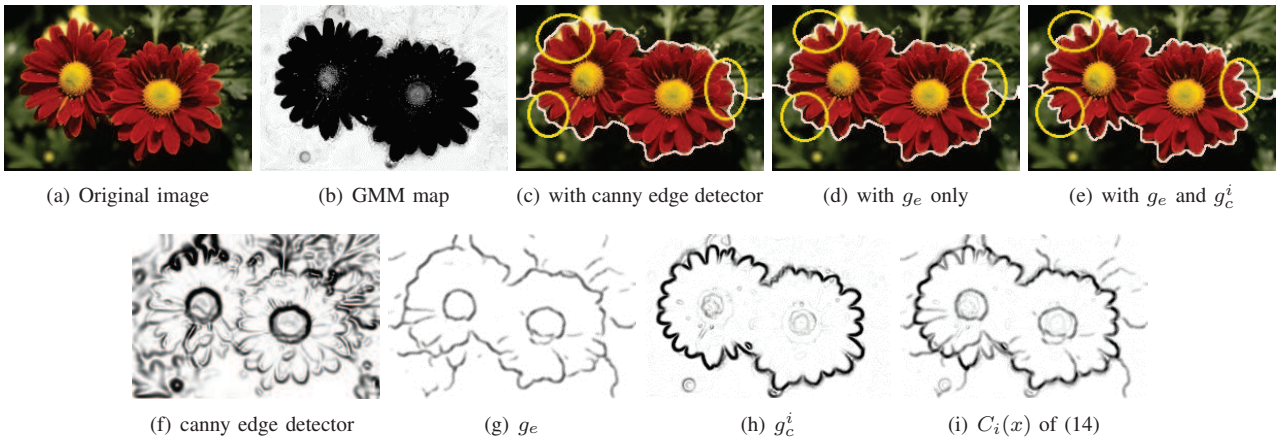
A. Test on Berkeley Benchmark

We have conducted experiment on the Berkeley segmentation dataset (BSDS) [9]. Particularly, we use the BSDS500 dataset that contains 500 images (300 images for training and 200 images for testing) and their corresponding human segmentations. The only parameter we calibrate over the training data is the tradeoff factor α in (1). In other words, we do not perform any training and α is empirically adjusted to a fix value ($\alpha = 0.1$) that achieves the best performance over the training images only. We evaluate the results using the



(a) Original image (b) RGB color (c) GMM (d) GMM+ GeoDesic (e) GMM+ GeoDesic+ Eigen

Fig. 3. The segmentation results by using different features. Combining global spectral, color and spatial features (e) achieves the best result.



(a) Original image (b) GMM map (c) with canny edge detector (d) with g_e only (e) with g_e and g_c^i

(f) canny edge detector (g) g_e (h) g_c^i (i) $C_i(x)$ of (14)

Fig. 4. Comparison of the results of our method using the two different $C_i(x)$ definitions in (3) and (14), respectively. Some boundary problems due to using (3) are circled in (c).

precision and recall framework of [21], where the *Precision* measures the proportion of how many machine generated boundaries can be found in human labelled boundaries and is sensitive to over-segmentation, while the *Recall* measures the proportion of how many human labelled boundaries can be found in machine generated boundaries and is sensitive to under-segmentation.

Table I lists the the precision and recall values of our method and some state-of-the-art algorithms including the gPb-owt-ucm method [9], Normalized Cut [1], Mean-Shift [7] and Graph Based Region Merging [2]. We also report the precision and recall values of the the EM method where the feature is a concatenation of RGB with eigenvectors and the original Potts model where only local color feature is used for comparison. The scores of the Potts model using different combinations of global features are also demonstrated. The performance gain is obvious with effective integration of the global information with the Potts model and the EM method in our algorithm. It can be seen that our algorithm obtains the highest precision with a value of 0.86 across the test dataset, which means most of the boundaries generated by our method match human segmentation. On the other hand, as our algorithm aims at producing sizable segments, it exhibits a certain degree of "under-segmentation" compared to other methods, and thus the recall value of our method is relatively lower. We would like to point out that the human segmentations offered in BSDS500 are of fine granularity, which goes against the goal of our algorithm, and thus the recall values do not fully reflect the performance of our method. We did experiment of removing some human segmentations with much finer granularity and recalculating the statistics of our results and obtained an increase in the recall score. Thus we believe that new metric and benchmark are needed to better evaluate methods for our task.

TABLE I
BOUNDARY BENCHMARKS ON BSDS500.

Method	Precision	Recall
EM method (RGB+eigenvectors)	0.46	0.49
Potts with RGB	0.53	0.43
Potts with GMM	0.6	0.49
Potts with GMM+Geodesic	0.72	0.54
Our Method	0.86	0.58
gPb-owt-ucm [9]	0.72	0.73
Mean Shift [7]	0.59	0.71
NCuts [1]	0.56	0.74
Felz-Hutt [2]	0.5	0.77

Our algorithm is implemented in C++ and runs on a Laptop with an Intel Core i7 1.73GHz Quad Core mobile processor, Nvidia Geforce GTX460M mobile graphics card and 8GB RAM. The average time to handle an image in the BSDS

is about 1∼3s with GPU acceleration.

B. Visual Results

Figure 5 shows some randomly selected images from the BSDS500 test dataset and their corresponding segmentation results using different algorithms. It can be seen that NCut often breaks smooth image regions since it requires a large input label number in order to obtain the correct boundaries. Felz-Hutt method usually produces many super-pixels, which causes severe visual artifacts. Blobworld and Mean-shift produce unpleasing segmentation results for complex images as they rely on local image features. As for the gPb-owt-ucm method, it can still produce trivial regions since it is constructed from region contours which can be of fine granularity. Compared with these existing methods, our proposed algorithm generates pleasing segmentation results with boundaries snapping to the geometry features of objects and a reasonable number of segments matching global human perception. Moreover, we also show the results of the original Potts model which only relies on local color feature. It can be observed that our proposed method that incorporates global color, spatial and structure information into the Potts model achieves much better visual results.

C. Limitations

Although the proposed method achieves very good visual results for most of the tested images, it still has some limitations. One limitation is that our method may ignore some small distinct regions due to the assumption of the method that the size of each segment is considerable. Another limitation is that for cluttered or camouflaged images that do not exhibit much structure information in eigenvectors, our algorithm does not perform well. We believe for such cases high-level knowledge should be involved in order to successfully segment the images.

V. Conclusion

This paper has described how to elaborately construct feature vectors for an image from low level cues resulted from the state-of-the-art techniques. The feature vectors consist of spectral attributes, global color and spatial information. The Potts model is formulated in terms of the feature vectors for segmentation. A heuristic approach is proposed to select the number of segments. As a result, a new algorithm is developed, which can automatically segment natural images into a small number of regions that are locally coherent, respect global structures, have smooth contours snapping to salient object boundaries, and correspond to meaningful objects. Experiments demonstrate that the proposed algorithm can achieve object-level segmentation to some extent.

Acknowledgment

The authors would like to acknowledge the Ph.D scholarship given by the Institute for Media Innovation, Nanyang Technological University, Singapore.

References

[1] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, 2000.

[2] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *International Journal of Computer Vision*, vol. 59, no. 2, pp. 167–181, 2004.

[3] C. Carson, S. Belongie, H. Greenspan, and J. Malik, "Blobworld: Image segmentation using expectation-maximization and its application to image querying," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 8, pp. 1026–1038, 2002.

[4] W. Yang, J. Cai, J. Zheng, and J. Luo, "User-friendly interactive image segmentation through unified combinatorial user inputs," *IEEE Transactions on Image Processing*, vol. 19, no. 9, pp. 2470–2479, 2010.

[5] J. Shotton, M. Johnson, and R. Cipolla, "Semantic texton forests for image categorization and segmentation," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, june 2008, pp. 1 –8.

[6] D. D. Hoffman and M. Singh, "Salience of visual parts," *Cognition*, vol. 63, no. 1, pp. 29 – 78, 1997. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0010027796007913

[7] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 5, pp. 603–619, 2002.

[8] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *NIPS*, T. G. Dietterich, S. Becker, and Z. Ghahramani, Eds. MIT Press, 2001, pp. 849–856.

[9] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898–916, 2011.

[10] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 11, pp. 1222–1239, 2001.

[11] D. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational problems," *Communications on Pure and Applied Mathematics*, vol. 42, no. 5, pp. 577–685, 1989. [Online]. Available: http://dx.doi.org/10.1002/cpa.3160420503

[12] T. Pock, T. Schoenemann, G. Graber, H. Bischof, and D. Cremers, "A convex formulation of continuous multi-label problems," in *ECCV (3)*, ser. Lecture Notes in Computer Science, D. A. Forsyth, P. H. S. Torr, and A. Zisserman, Eds., vol. 5304. Springer, 2008, pp. 792–805.

[13] E. Bae, J. Yuan, X. C. Tai, and Y. Boykov, "A fast continuous max-flow approach to non-convex multilabeling problems," *SIAM Imag. Sciences*, 2010.

[14] J. Yuan, E. Bae, X.-C. Tai, and Y. Boykov, "A continuous max-flow approach to potts model," in *ECCV (6)*, ser. Lecture Notes in Computer Science, K. Daniilidis, P. Maragos, and N. Paragios, Eds., vol. 6316. Springer, 2010, pp. 379–392.

[15] R. B. Potts, "Some Generalized Order-Disorder Transformation," in *Transformations, Proceedings of the Cambridge Philosophical Society*, vol. 48, 1952, pp. 106–109.

[16] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Transactions on Image Processing*, vol. 10, no. 2, pp. 266–277, 2001.

[17] T. Pock, A. Chambolle, D. Cremers, and H. Bischof, "A convex relaxation approach for computing minimal partitions," in *CVPR*. IEEE, 2009, pp. 810–817.

[18] E. Bae, J. Yuan, and X.-C. Tai, "Global minimization for continuous multiphase partitioning problems using a dual approach," *International Journal of Computer Vision*, vol. 92, no. 1, pp. 112–129, 2011.

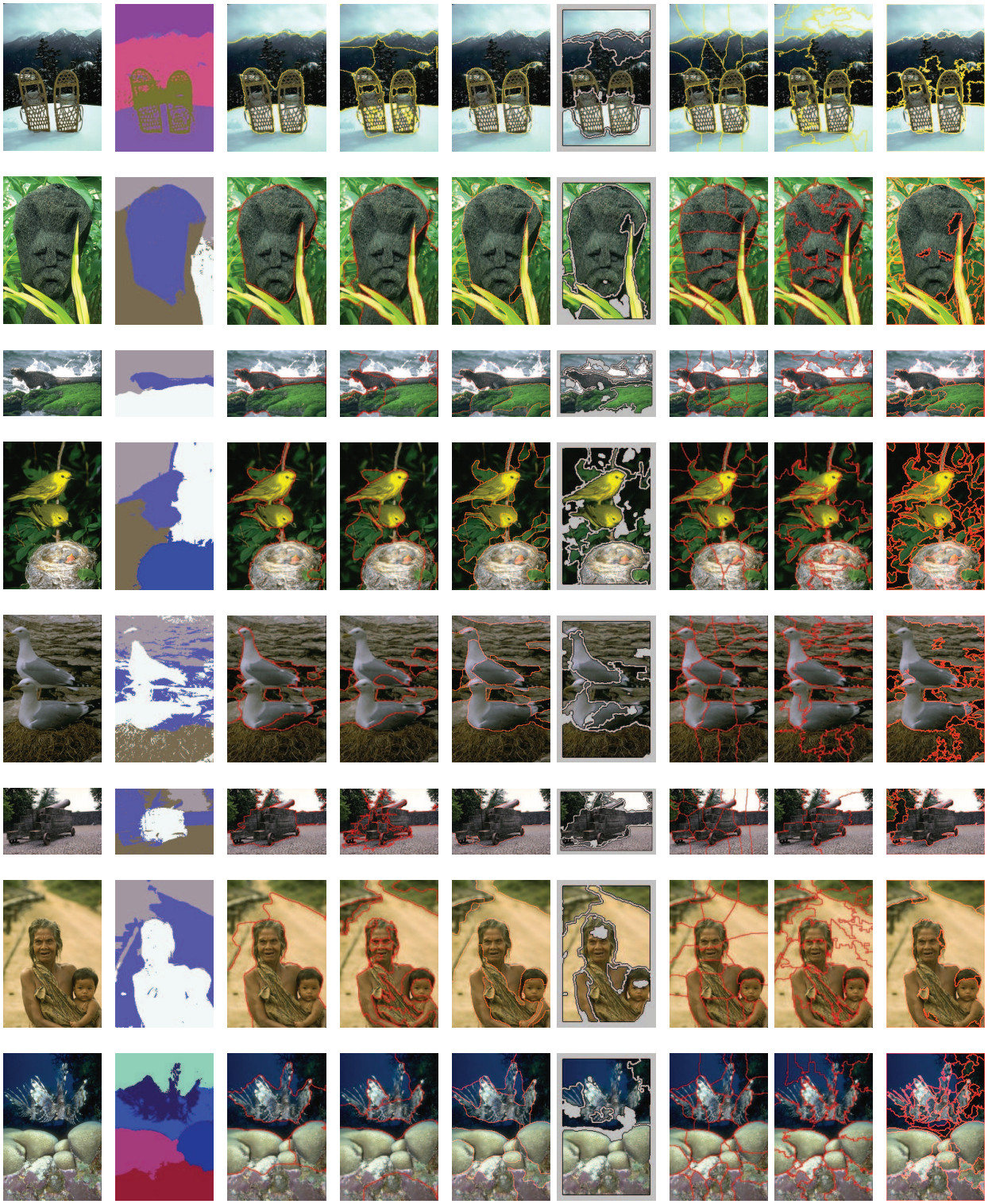
[19] J. Lellmann, J. H. Kappes, J. Yuan, F. Becker, and C. Schnörr, "Convex multi-class image labeling by simplex-constrained total variation," in *SSVM*, ser. Lecture Notes in Computer Science, X.-C. Tai, K. Mørken, M. Lysaker, and K.-A. Lie, Eds., vol. 5567. Springer, 2009, pp. 150–162.

[20] C. Zach, D. Gallup, J.-M. Frahm, and M. Niethammer, "Fast global labeling for real-time stereo using multiple plane sweeps," in *VMV*, O. Deussen, D. A. Keim, and D. Saupe, Eds. Aka GmbH, 2008, pp. 243–252.

[21] D. R. Martin, C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 5, pp. 530–549, 2004.

[22] U. von Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007.

[23] C. Rother, V. Kolmogorov, and A. Blake, "GrabCut: interactive foreground extraction using iterated graph cuts," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 309–314, 2004.



(a) input images (b) EM initialization (c) Our method (d) Potts with RGB (e) gPb-owt-ucm (f) Blobworld (g) NCUT (h) Felz-Hutt (i) Meanshift

Fig. 5. Examples of some images randomly selected from the BSDS500 test dataset and their corresponding segmentation results using different methods with optimal parameters tuned over the training set. Note that red and yellow contours depict the region boundaries. For Blobworld, it uses white contours for boundaries and the gray regions indicate unlabelled pixels.

[24] X. Bai and G. Sapiro, "A geodesic framework for fast interactive image and video segmentation and matting," in *ICCV*. IEEE, 2007, pp. 1–8.
[25] J. Malik, S. Belongie, J. Shi, and T. K. Leung, "Textons, contours and regions: Cue integration in image segmentation," in *ICCV*, 1999, pp. 918–925.