# Iterative MAP and ML Estimations for Image Segmentation

Shifeng Chen[1], Liangliang Cao[2], Jianzhuang Liu[1], and Xiaoou Tang[1,3]
[1]Dept. of IE, The Chinese University of Hong Kong
{sfchen5, jzliu}@ie.cuhk.edu.hk
[2]Dept. of ECE, University of Illinois at Urbana-Champaign
cao4@uiuc.edu
[3]Microsoft Research Asia, Beijing, China
xitang@microsoft.com

## Abstract

*Image segmentation plays an important role in computer vision and image analysis. In this paper, the segmentation problem is formulated as a labeling problem under a probability maximization framework. To estimate the label configuration, an iterative optimization scheme is proposed to alternately carry out the maximum a posteriori (MAP) estimation and the maximum-likelihood (ML) estimation. The MAP estimation problem is modeled with Markov random fields (MRFs). A graph-cut algorithm is used to find the solution to the MAP-MRF estimation. The ML estimation is achieved by finding the means of region features. Our algorithm can automatically segment an image into regions with relevant textures or colors without the need to know the number of regions in advance. In addition, under the same framework, it can be extended to another algorithm that extracts objects of a particular class from a group of images. Extensive experiments have shown the effectiveness of our approach.*

## 1. Introduction

Image segmentation has received extensive attention since the early years of computer vision research. Due to the limitation of computational ability, the early segmentation methods [12], [15] flaw in efficiency and/or performance.

Recently, Shi and Malik proposed to apply normalized cuts to image segmentation [13], [16], which is able to capture intuitively salient parts in an image. The normalized cuts has an important advantage in spectral clustering. However, it is not perfectly fit for the nature of image segmentation because ad hoc approximations must be introduced to relax the NP-hard computational problem. These approximations are not well understood and often lead to unsatisfactory results.

Expectation-maximization (EM) [4] is another interesting segmentation method. One shortcoming of EM is that the number of regions is kept unchanged during the segmentation, which often causes wrong results because different images usually have different numbers of regions. Theoretically, people can use the minimum description length (MDL) principle [4] to alleviate this problem, but the segmentation has to be carried out many times with different region numbers to find the best result. This takes a large amount of computation, and the theoretically best result may not accord with our perception.

Tu and Zhu [14] presented an generative segmentation method under the framework of MAP estimation of MRFs, with the Markov Chain Monte Carlo (MCMC) used to solve the MAP-MRF estimation. This method suffers much from the computation burden. In addition, the generative approach explicitly models regions in images with many constraints, resulting in the difficulty of choosing parameters to express objects in images. Another popular segmentation approach based on MRFs is graph cut algorithms. The algorithms in [2] and [8] rely on human interaction, and solve the two-class segmentation problem. In [17], Zabih and Kolmogorov used graph cuts to obtain the segmentation of multiple regions in an image, but the number of clusters is given in the beginning and cannot be adjusted during the segmentation. Besides, the segmentation result is sensitive to this number, as pointed out by the authors.

Recently, some researchers start to pay attention to learning-based segmentation of objects of a particular class from images [9], [1], [10], [11]. Different from common segmentation methods, this work requires to learn the parameters of a model expressing the same objects (say, horse) from a set of images.

This paper proposed a new image segmentation algorithm based on a probability maximization model. An iterative optimization scheme alternately making the MAP and ML estimations is the key to the segmentation. We model the MAP estimation with MRFs and solve the MAP-MRF estimation problem using graph cuts. The ML estimation is obtained by finding the means of region features.

1-4244-1180-7/07/$25.00 ©2007 IEEE

The contributions of this work include: 1) a novel probabilistic model and an iterative optimization scheme for image segmentation; 2) using graph cuts to solve the multiple region segmentation problem with the number of regions automatically adjusted according to the properties of the regions; and 3) extracting objects from a group of images containing the objects of the same class.

## 2. A New Probabilistic Model

For a given image $P$, the features of every pixel $p$ are expressed by a four-dimensional vector

$$\mathbf{I}(p) = (I_L(p), I_a(p), I_b(p), I_t(p))^T, \tag{1}$$

where $I_L(p)$, $I_a(p)$ and $I_b(p)$ are the components of $p$ in the L*a*b* color space and $I_t(p)$ denotes the texture feature of $p$. Several classical texture descriptors have been developed in [4], [6], and, [7]. In this paper, the texture contrast defined in [4] (scaled from $[0, 1]$ to $[0, 255]$) is chosen as the texture descriptor.

The task of image segmentation is to group the pixels of the image into relevant regions. If we formulate it as a labeling problem, the objective is then to find a label configuration $f = \{f_p \mid p\}$ where $f_p$ is the label of pixel $p$ denoting which region this pixel is grouped into. Suppose that we have $k$ possible region labels. A four-dimensional vector

$$\boldsymbol{\phi}(i) = (\overline{I}_L(i), \overline{I}_a(i), \overline{I}_b(i), \overline{I}_t(i))^T \tag{2}$$

is used to describe the properties of label (region) $i$, where the four components of $\boldsymbol{\phi}(i)$ have the similar meanings to those of the corresponding four components of $\mathbf{I}(p)$.

Let $\Phi = \{\boldsymbol{\phi}(i)\}$ be the union of the region features. If $P$ and $\Phi$ are known, the segmentation is to find an optimal label configuration $\widehat{f}$, which maximizes the posterior possibility of the label configuration:

$$\widehat{f} = \arg\max_f \mathbf{Pr}(f|\Phi, P), \tag{3}$$

where $\Phi$ can be obtained by either a learning process or an initialized estimation. However, due to the existence of noise and diverse objects in different images, it is difficult to obtain $\Phi$ that is precise enough. Our strategy here is to refine $\Phi$ according to the current label configuration found by (3). Thus, we propose to use an iterative method to solve the segmentation problem.

Suppose that $\Phi^n$ and $f^n$ are the estimation results in the $n$th iteration. Then the iterative formulas for optimization are

$$f^{n+1} = \arg\max_f \mathbf{Pr}(f|\Phi^n, P), \tag{4}$$

$$\Phi^{n+1} = \arg\max_\Phi \mathbf{Pr}(f^{n+1}|\Phi, P). \tag{5}$$

This iterative optimization is preferred because (4) can be solved by the MAP estimation, and (5) by the ML estimation.

### 2.1. MAP Estimation of $f$ from $\Phi$

Given an image $P$ and the potential region features $\Phi$, $\mathbf{Pr}(f|\Phi, P)$ can be obtained by the Bayesian law:

$$\mathbf{Pr}(f|\Phi, P) \propto \mathbf{Pr}(\Phi, P|f)\mathbf{Pr}(f), \tag{6}$$

which is a MAP estimation problem and can be modelled using MRFs.

Assuming that the observation of the image follows an independent identical distribution (i.i.d.), we define

$$\mathbf{Pr}(\Phi, P|f) \propto \prod_{p\in P} \exp\left(-D(p, f_p, \Phi)\right), \tag{7}$$

where $D(p, f_p, \Phi)$ is the data penalty function which imposes the penalty of a pixel $p$ with a label $f_p$ for given $\Phi$. The data penalty function is defined as:

$$D(p, f_p, \Phi) = ||\mathbf{I}(p) - \boldsymbol{\phi}(f_p)||^2. \tag{8}$$

We restrict our attention to MRFs whose clique potentials involve pairs of neighboring pixels. Thus

$$\mathbf{Pr}(f) \propto \exp\left(-\sum_{p\in P}\sum_{q\in\mathcal{N}(p)} V_{p,q}(f_p, f_q)\right), \tag{9}$$

where $\mathcal{N}(p)$ is the neighborhood of pixel $p$. $V_{p,q}(f_p, f_q)$, called the smoothness penalty function, describes the prior probability of a particular label configuration with the elements of the clique $(p, q)$. It is defined using a generalized Potts model [3]:

$$V_{p,q}(f_p, f_q) = c \cdot \exp\left(\frac{-\Delta(p,q)}{\sigma}\right) \cdot T(f_p \neq f_q), \tag{10}$$

where $\Delta(p, q) = |I_L(p) - I_L(q)|$ denotes how different the brightnesses of $p$ and $q$ are, $c > 0$ is a smoothness factor, $\sigma > 0$ is used to control the contribution of $\Delta(p, q)$ to the penalty, and $T(\cdot)$ is 1 if its argument is true and 0 otherwise. $V_{p,q}(f_p, f_q)$ depicts two kinds of constraints. The first enforces the spatial smoothness; if two neighboring pixels are labelled differently, a penalty is imposed. The second considers a possible edge between $p$ and $q$; if two neighboring pixels cause a larger $\Delta$, then they have greater likelihood to be partitioned into two regions.

From (6), (7), and (9), we have

$$\mathbf{Pr}(f|\Phi, P) \propto \left(\prod_{p\in P} \exp\left(-D(p, f_p, \Phi)\right)\right) \cdot \exp\left(-\sum_{p\in P}\sum_{q\in\mathcal{N}(p)} V_{p,q}(f_p, f_q)\right). \tag{11}$$

Taking the logarithm of (11), we have the energy function:

$$E(f, \Phi) = \sum_{p\in P}\left(D(p, f_p, \Phi) + \sum_{q\in\mathcal{N}(p)} V_{p,q}(f_p, f_q)\right). \tag{12}$$

It includes two parts: the data term

$$E_{data} = \sum_{p \in P} D(p, f_p, \Phi) \tag{13}$$

and the smoothness term

$$E_{smooth} = \sum_{p \in P} \sum_{q \in \mathcal{N}(p)} V_{p,q}(f_p, f_q). \tag{14}$$

From (12), we see that maximizing $\mathbf{Pr}(f|\Phi, P)$ is equivalent to minimizing the Markov energy $E(f, \Phi)$ for given $\Phi$. In this paper, we use a graph cut algorithm to solve this minimization problem, which is described in Section 3.

### 2.2. ML Estimation of $\Phi$ from $f$

If the label configuration $f$ is given, the optimal $\Phi$ should maximize $\mathbf{Pr}(f|\Phi, P)$, or minimize $E(f, \Phi)$ equivalently. Thus we have

$$\nabla_\Phi \log \mathbf{Pr}(f|\Phi, P) = \mathbf{0}, \text{ or } \nabla_\Phi E(f, \Phi) = \mathbf{0}, \tag{15}$$

where $\nabla_\Phi$ denotes the gradient operator. Since $V_{p,q}(f_p, f_q)$ is independent of $\Phi$, we obtain

$$\nabla_\Phi \sum_{p \in P} D(p, f_p, \Phi) = \mathbf{0}, \tag{16}$$

where different formulations of $D(p, f_p, \Phi)$ lead to different estimations of $\Phi$. For our formulation in (8), it follows that

$$\sum_{p \in P} D(p, f_p, \Phi) = \sum_i \sum_{f_p = i} ||\mathbf{I}(p) - \boldsymbol{\phi}(i)||^2. \tag{17}$$

From (16) and (17), we obtain the ML estimation $\Phi = \boldsymbol{\phi}(i)$, where

$$\boldsymbol{\phi}(i) = \frac{1}{num_i} \sum_{f_p = i} \mathbf{I}(p), \tag{18}$$

with $num_i$ being the number of pixels within region $i$. Here (18) is exactly the equation to obtain $\overline{I}_L(i), \overline{I}_a(i), \overline{I}_b(i)$, and $\overline{I}_t(i)$ in (2).

Note that when the label configuration $f = \{f_p|p\}$ is unknown, finding the solution of (16) is carried out by clustering the pixels into groups. In the case, the ML estimation is achieved by the $K$-means algorithm [5], which serves as the initialization in the algorithm described in Section 3.

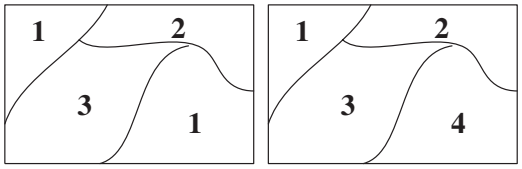
## 3. The Proposed Algorithm

With $E(f, \Phi)$ defined in (12), the estimation of $\widehat{f}$ and $\widehat{\Phi}$ in (4) and (5) are now transformed to

$$f^{n+1} = \arg\min_f E(f, \Phi^n), \tag{19}$$

$$\Phi^{n+1} = \arg\min_\Phi E(f^{n+1}, \Phi). \tag{20}$$

The two equations correspond to the MAP estimation and the ML estimation, respectively. The algorithm to obtain $\widehat{f}$ and $\widehat{\Phi}$ is described as Algorithm 1.



(a) (b)

Figure 1. Relabeling of the regions. (a) The result before the relabeling. (b) The result after the relabeling.

**Algorithm 1:** Our segmentation algorithm.

**Input**: an RGB color image.
- **Step 1:** Convert the image into L*a*b* space and calculate the texture contrast.
- **Step 2:** Use the $K$-means algorithm to initialize $\Phi$.
- **Step 3:** Iterative optimization.
  - **3.1:** MAP estimation — Estimate the label configuration $f$ based on current $\Phi$ using the graph cut algorithm [3].
  - **3.2:** Relabeling — Set a unique label to each connecting region to form a new cluster, obtaining a new $f$.
  - **3.3:** ML estimation — Refine $\Phi$ based on current $f$ with (18).
- **Step 4:** If $\Phi$ and $f$ do not change between two successive iterations or the maximum number of iterations is reached, go to the output step; otherwise, go to step 3.

**Output**: Multiple segmented regions.

We explain step 3.2 in more details here. After step 3.1, it is possible that two non-adjacent regions are given the same label (see Fig. 1(a)). After step 3.2, each of the connected regions has a unique label (see Fig. 1(b)).

One remarkable feature of our algorithm is the ability to adjust the region number automatically during the iterative optimization by the relabeling step. Fig. 2 gives an example to show how the iterations improve the segmentation results. Comparing Figs. 2(b), (c), and (d), we can see that the final result is the best.

## 4. Object Extraction from a Group of Images

The framework proposed in Section 2 can be extended from single image segmentation to object extraction from a group of images. These images contain objects of the same class with similar colors and textures. The purpose of the specific algorithm developed next is not to segment an image into multiple regions, but to extract interested objects from a group of images containing these similar objects. Unlike the learning-based segmentation of a particular class of objects [9], [1], [10], [11], our algorithm does not need to learn a deformable shape model representing the objects. Instead, we assume that one pixel is known which is inside an interested object in one (only one) image of the group. In our current experiments, this pixel is provided by the user

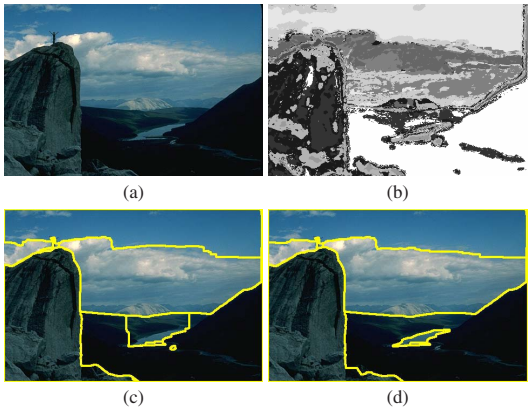


Figure 2. A segmentation example. (a) An original image. (b) The result of initial $K$-means clustering with $K = 10$. (c) The result of the first iteration with $K$ adjusted to 8. (d) The converged result after 4 iterations with $K$ changed to 6.

clicking once at the object.

Given a group of $N$ images containing a class of objects and the known pixels' position, our algorithm estimates the features of the interested objects in the group and combines the estimated features into the same framework of the ML and MAP estimations to extract the objects from the images.

To estimate the features of the interested objects, we model them as a Gaussian distribution with a mean vector $\boldsymbol{\phi}^*$ and a variance matrix $\Sigma$. Using Algorithm 1, we can segment each image into $n_k$ regions, $1 \le k \le N$. In every image, the features of each region are represented by $\boldsymbol{\phi}_k(i)$, $1 \le i \le n_k$. Now we need to estimate $\boldsymbol{\phi}^*$ and $\Sigma$ from the images.

We use an iterative ML estimation to find the Gaussian model of the objects $(\boldsymbol{\phi}^*, \Sigma)$. At first, we initialize $\boldsymbol{\phi}^*$ as the features of the region $R$ containing the known pixel in the image, and set $\Sigma$ as an identity matrix. Then the algorithm selects one region that is most similar to $R$ from each image. These selected regions are used to perform the ML estimation of $\boldsymbol{\phi}^*$ and $\Sigma$. The two iterative steps for the estimation are described as follows:

**Step 1:** For each image $k$,

$$\boldsymbol{\phi}_k^* = \arg\max_{\boldsymbol{\phi}_k(i)} \mathbf{Pr}(\boldsymbol{\phi}_k(i)|\boldsymbol{\phi}^*, \Sigma), \tag{21}$$

where

$$\mathbf{Pr}(\boldsymbol{\phi}_k(i)|\boldsymbol{\phi}^*, \Sigma) \propto \frac{1}{|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\boldsymbol{\phi}_k(i) - \boldsymbol{\phi}^*)^T \Sigma^{-1} (\boldsymbol{\phi}_k(i) - \boldsymbol{\phi}^*)\right).$$

**Step 2:** $$(\boldsymbol{\phi}^*, \Sigma) = \arg\max_{\boldsymbol{\phi}^*, \Sigma} \prod_k \mathbf{Pr}(\boldsymbol{\phi}_k^*|\boldsymbol{\phi}^*, \Sigma). \tag{22}$$

After $\boldsymbol{\phi}^*$ is found, we set the object feature vector $\boldsymbol{\phi}_O = \boldsymbol{\phi}^*$. On the other hand, for each image, $m_B$ regions' feature vectors (called the background feature vectors) which are farthest from $\boldsymbol{\phi}^*$ are used to form a set $\boldsymbol{\phi}_B$. Extracting an object out of an image $k$ is to set a binary value $f_p = 0$ or 1 to each pixel $p$ in image $k$, where $f_p = 0$ (or 1) denotes pixel $p$ belongs to the background (or object). This task may be thought of as a two-class segmentation problem that can be solved using the MAP-ML estimation presented in Section 3. Here, we define a new data penalty function

$$D(p, f_p = 1) = ||\mathbf{I}(p) - \boldsymbol{\phi}_O||^2, \tag{23}$$

$$D(p, f_p = 0) = \arg\min_{\overline{\boldsymbol{\phi}}_B \in \boldsymbol{\phi}_B} ||\mathbf{I}(p) - \overline{\boldsymbol{\phi}}_B||^2. \tag{24}$$

When $\boldsymbol{\phi}_O$ is known, we can obtain $f$ by the MAP estimation via graph cuts to minimize

$$E(f, \boldsymbol{\phi}_O) = \sum_{p \in P} D(p, f_p) + \sum_{p \in P} \sum_{q \in \mathcal{N}(p)} V_{p,q}(f_p, f_q), \tag{25}$$

where $D(p, f_p)$ is defined in (23) and (24). Since the initialized $\boldsymbol{\phi}_O$ is generally a rough estimation, it is necessary to update it with the ML estimation, as described in Section 3. Then this MAP-ML estimation is repeated again until it converges. The summary of the algorithm for object extraction from one group of images is described in Algorithm 2.

**Algorithm 2**: Object extraction from a group of images.

**Input:** $N$ images containing the same class of objects.

**Step 1:** Click the object in one arbitrary image, and record the clicked pixel as $p_0$.

**Step 2:** Segment each image into $n_k$ regions using Algorithm 1, $1 \le k \le N$.

**Step 3:** Learn the object features $\boldsymbol{\phi}^*$ and $\Sigma$ using (21) and (22) with a general ML estimation algorithm with the initial $\boldsymbol{\phi}^*$ from the region containing $p_0$ and the identity matrix $\Sigma$.

**Step 4:** Extract the objects from each image:

**4.1:** Initialization — Set $\boldsymbol{\phi}_O$ to $\boldsymbol{\phi}^*$.

**4.2**: MAP estimation — Perform the MAP estimation of the label configuration $f$ via graph cuts.

**4.3:** ML estimation — Refine $\boldsymbol{\phi}_O$ based on current $f$ with (18).

**Step 5:** Perform Steps 4.2 and 4.3 iteratively until $E$ converges or the maximum iteration number is reached.

**Output:** The extracted objects from the images.

## 5. Experimental Results

We test the proposed Algorithm 1 with a large set of natural images and compare the results with those obtained by the normalized cuts [13] and the blobworld [4]. The normalized cuts is a popular segmentation algorithm, and the blobworld is recently published using the EM framework. In our algorithm, we set the initial cluster number in the $K$-means algorithm to 10. The region number in the normalized cuts is set to 10. The cluster number in the blobworld

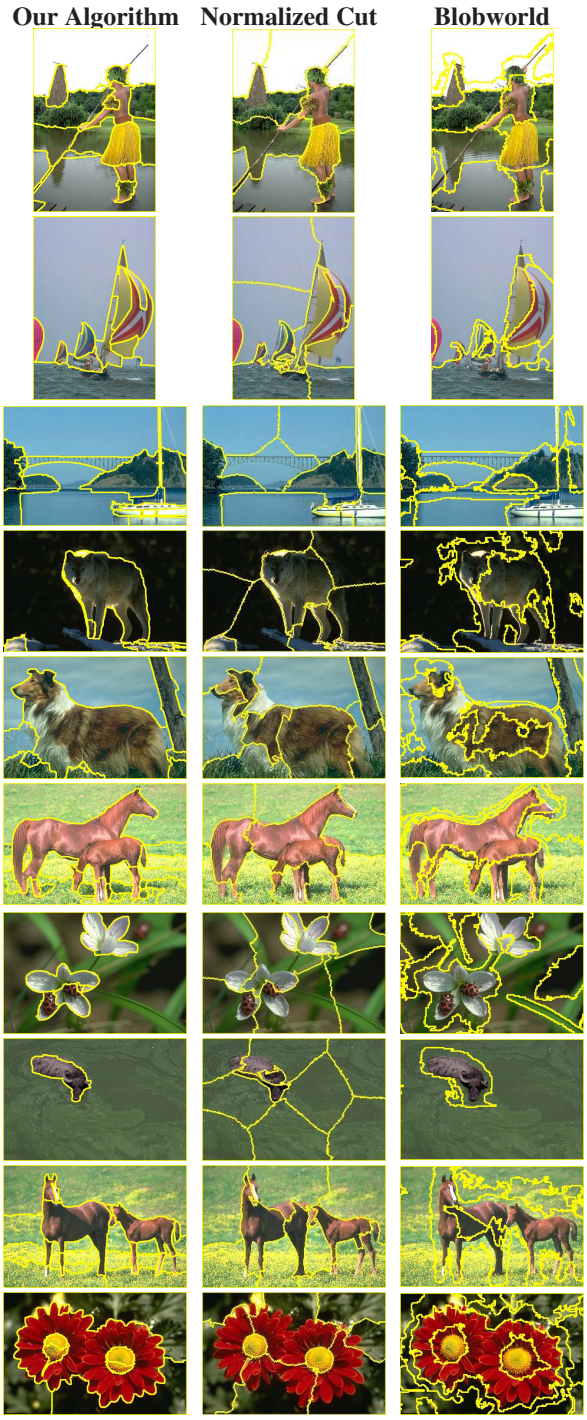



Figure 3. Results obtained by the three algorithms for single image segmentation. The boundaries of the segmentation results are marked using yellow curves.

is initialized as 3, 4, and 5, and then the MDL is used to choose the best one, which is suggested in [4].

Some of the experimental results are shown in Fig. 3, from which we can see that our algorithm outperforms the other two. First, the found edges in our results match the real edges much better. The normalized cut algorithm tends to partition an image into large regions, making the boundaries apart from the real edges. On the other hand, since the blobworld integrates no edge information, the boundaries of its results are far from satisfactory with too many rough and trivial edges. Second, our algorithm can adapt the region number to different images automatically although all the initial region numbers set for the $K$-means algorithm are 10.

To test our Algorithm 2, we try three groups of images, bulls, horses, and zebras, in the experiment. The results are given in Fig. 4. Each group has 8 images. The size and posture of these animals vary significantly, and their legs in some images are occluded by the grasses. Our algorithm is able to extract them under these nontrivial conditions. Note that we need neither a large set of images for training nor predefined templates in order to extract the objects.

## 6. Conclusions

In this paper, we have developed two novel algorithms under the same framework. Algorithm 1 is for single image segmentation and Algorithm 2 for object extraction from a group of images. Our algorithms are formulated as a labeling problem using a probability maximization model. An iterative optimization technique combining the MAP-MRFs and ML estimations is employed in our framework. The MAP-MRFs problem is solved using graph cuts and the ML estimation is obtained by finding the means of the region features. We have compared our Algorithm 1 with the normalized cuts and the blobworld. The experimental results show that our algorithm outperforms the other two.

One of our future works aims to generalize the extraction of objects with homogeneous region features to the extraction of objects composed of two or more regions with different features, such as humans and cars. One possible way is to design complex models to describe the region features of the objects, and consider the image context for the object extraction.

## Acknowledgement

This work was supported by the Research Grants Council of the Hong Kong SAR (Project No. CUHK 414306) and the CUHK Direct Grant.

## References

[1] E. Borenstein and S. Ullman. Learning to segment. In *ECCV*, 2004. 1, 3

[2] Y. Boykov and M. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images. In *ICCV*, 2001. 1

[3] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Trans. PAMI*, 23(11):1222–1239, 2001. 2, 3

[4] C. Carson, S. Belongie, H. Greenspan, and J. Malik. Blobworld: Image segmentation using expectation-maximization

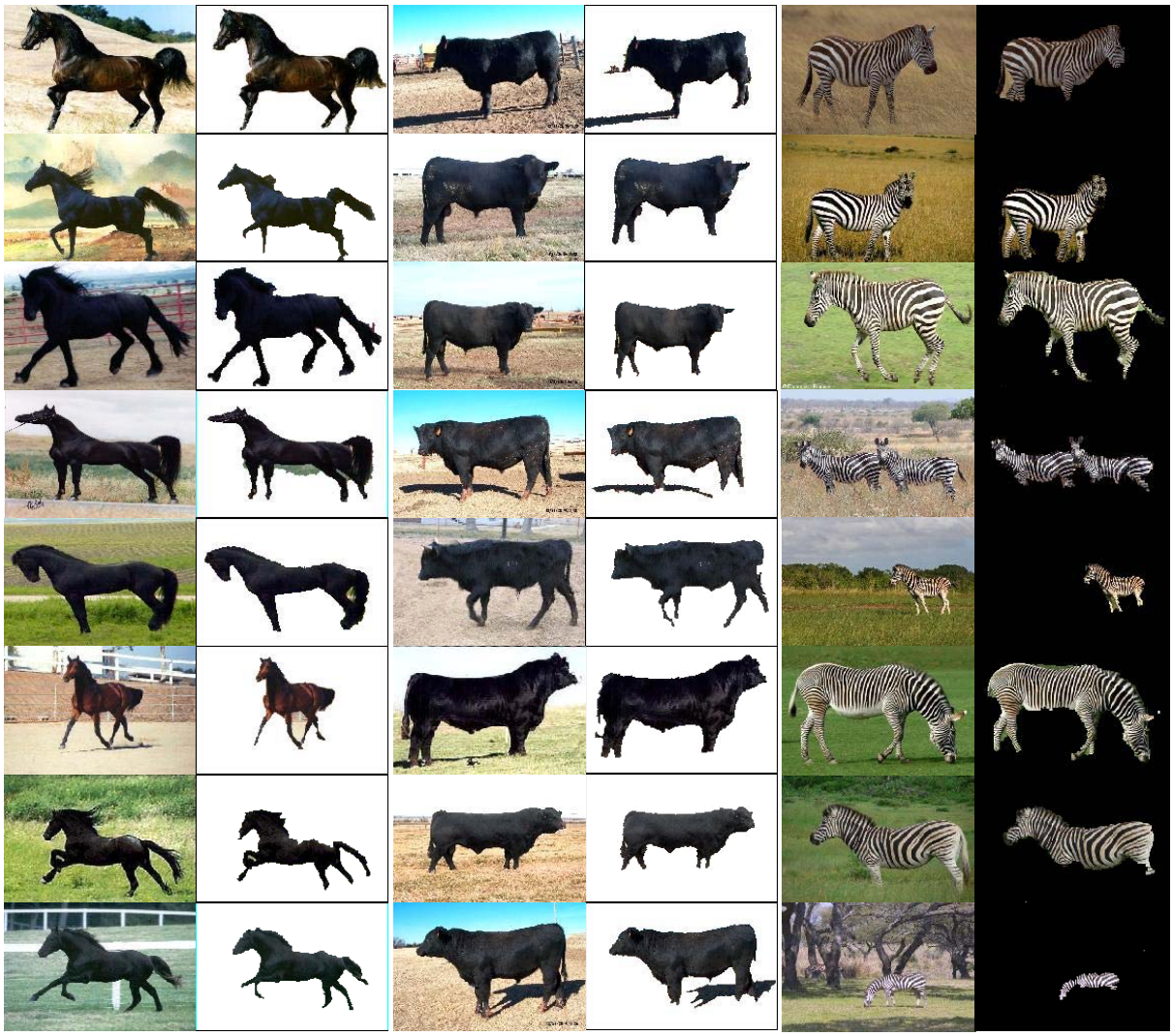


Figure 4. Results of our algorithm for the extraction of objects from two groups of images. The first, third, and fifth column include 8 bull images, horse images and zebra images respectively. The second, fourth, and sixth column are the extracted animals.


and its application to image querying. *IEEE Trans. PAMI*, 24(8):1026–1038, 2002. 1, 2, 4, 5

[5] R. Duda, P. Hart, and D. Stork. *Pattern Classification*. Wiley-Interscience, 2 edition, 2001. 3

[6] W. Förstner. A framework for low level feature extraction. In *ECCV*, 1994. 2

[7] J. Gårding and T. Lindeberg. Direct computation of shape cues using scale-adapted spatial derivative operators. *IJCV*, 17(2):163–191, 1996. 2

[8] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *IEEE Trans. PAMI*, 26(2):147–159, 2004. 1

[9] M. P. Kumar, P. H. S. Torr, and A. Zisserman. Obj cut. *CVPR*, 2005. 1, 3

[10] A. Opelt, A. Pinz, and A. Zisserman. A boundary-fragment-model for object detection. In *ECCV*, 2006. 1, 3

[11] A. Opelt, A. Pinz, and A. Zisserman. Incremental learning of object detectors using a visual shape alphabet. In *ICCV*, 2006. 1, 3

[12] T. Pappas. An adaptive clustering algorithm for image segmentation. *IEEE Trans. Signal Processing*, 40(4):901–914, 1992. 1

[13] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Trans. PAMI*, 22(8):888–905, 2000. 1, 4

[14] Z. Tu and S.-C. Zhu. Image segmentation by data-driven markov chain monte carlo. *IEEE Trans. PAMI*, 24(5):657–673, 2002. 1

[15] L. Vincent and P. Soille. Watersheds in digital spaces: An efficient algorithm based on immersion simulations. *IEEE Trans. PAMI*, 13(6):583–598, 1991. 1

[16] S. X. Yu and J. Shi. Multiclass spectral clustering. In *ICCV*, 2003. 1

[17] R. Zabih and V. Kolmogorov. Spatially coherent clustering using graph cuts. In *CVPR*, 2004. 1