International Journal of Modern Engineering Research (IJMER)
www.ijmer.com Vol.2, Issue.4, July-Aug. 2012 pp-2635-2640 ISSN: 2249-6645

A Extensive Technique For Image Segmentation By Using Algorithm Of MAP-ML

Vijayadheeswar Reddy
M.Tech Student Scholar, DIP,
Dept of Electronics and Communication Engineering,
Nalanda Institute of Engineering and technology,
Sattenapalli (M); Guntur (Dt); A.P, India

I. V. G. Manohar
M.Tech , Asst Professor
Dept of Electronics and Communication Engineering,
Nalanda Institute of Engineering and technology,
Sattenapalli (M); Guntur (Dt); A.P, India.

Abstract: *Image segmentation is the process of dividing an image into multiple parts. This is typically used to identify objects or other relevant information in digital images. maximum a posteriori probability (MAP) estimate is a mode of the posterior distribution. The MAP can be used to obtain a point estimate of an unobserved quantity on the basis of empirical data. It is closely related to Fisher's method of maximum likelihood (ML), but employs an augmented optimization objective which incorporates a prior distribution over the quantity one wants to estimate. MAP estimation can therefore be seen as a regularization of ML estimation. In this Paper our algorithm segment an image into regions with relevant Textures automatically there is no need of regions. The simulations shows the image edges very well. Comparing to six state-of-the-art algorithms, extensive experiments have shown that our algorithm performs the best.*

Keywords: *Image segmentation, graph cuts, maximum likelihood, maximum a posteriori, Markov random fields.*

I. Introduction

A central problem, called *segmentation*, is to distinguish objects from background . For intensity images (ie, those represented by point-wise intensity levels) four popular approaches are: threshold techniques, edge-based methods, region-based techniques, and connectivity-preserving relaxation methods. Threshold techniques, which make decisions based on local pixel information, are effective when the intensity levels of the objects fall squarely outside the range of levels in the background. Because spatial information is ignored, however, blurred region boundaries can create havoc.

Edge-based methods center around contour detection: their weakness in connecting together broken contour lines make them, too, prone to failure in the presence of blurring. A region-based method usually proceeds as follows: the image is partitioned into connected regions by grouping neighboring pixels of similar intensity levels. Adjacent regions are then merged under some criterion involving perhaps homogeneity or sharpness of region boundaries. Over stringent criteria create fragmentation; lenient ones overlook blurred boundaries and over merge. Hybrid techniques using a mix of the methods above are also popular.

A connectivity-preserving relaxation-based segmentation method, usually referred to as the *active contour model*, was proposed recently. The main idea is to start with some initial boundary shape represented in the form of spline curves, and iteratively modify it by applying various shrink/expansion operations according to some energy function. Although the energy-minimizing model is not new, coupling it with the maintenance of an ``elastic'' contour model gives it an interesting new twist. As usual with such methods, getting trapped into a local minimum is a risk against which one must guard; this is no easy task. In contrast to the heuristic nature of these approaches, computational geometry suggests a more algorithmic tack. One would first formalize a mathematical criterion for the ``goodness'' of a given segmentation. This would allow us to formulate the segmentation problem as an optimization problem under certain geometric constraints.

The problem of image segmentation and visual grouping has received extensive attention since the early years of computer vision research. It has been known that visual grouping plays an important role in human visual perception. Many computer vision problems, such as stereo vision, motion estimation, image retrieval, and object recognition, can be solved better with reliable results of image segmentation. For example, results of stereo vision based on image segmentation are more stable than pixel-based results. Although the problem of image segmentation has been studied for more than three decades, great challenges still remain in this research. Here we presented to apply normalized cuts to image segmentation [1] 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.

These approaches are physics-based models that deform under the laws of Newton mechanics, in particular, by the theory of elasticity expressed in the Lagrange dynamics. Many contour based segmentation algorithms[2]-[4] have been developed in the past two decades. One problem existing in these algorithms is that they are easy to get trapped in local minima. In addition, they need manually specified initial curves close to the objects of interest. Region-based approaches try to classify an image into multiple consistent regions or classes. Thresholding is the simplest segmentation method but its performance is usually far from satisfactory. Watershed segmentation [10], [11] is one of the traditional region-based approaches. The watershed transform is often used to segment touching objects. It finds intensity valleys in an image if the image is viewed as a surface with mountains (high intensity regions) and valleys (low intensity regions). Morphological operations are always used to handle the over-segmented problem in the output obtained by the watershed transform. Usually, watershed is used for the segmentation of foreground and background (two class) of an image. For a general color image with many different regions, it often gives a bad result. It is also sensitive to the morphological structuring element.

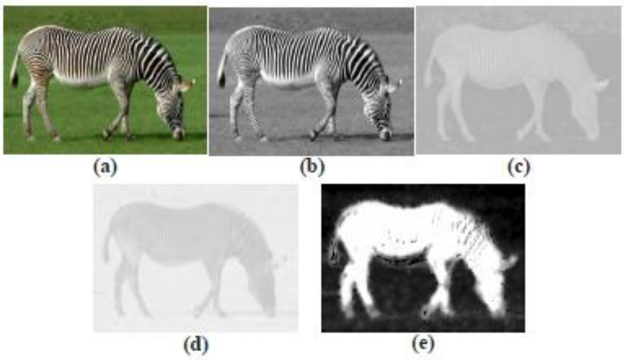
This paper proposes a new image segmentation algorithm based on a probability maximization model. An iterative optimization scheme alternately making the MAP and the maximum likelihood (ML) estimations is the key to the segmentation. We model the MAP estimation with MRFs and solve the MAP-MRF estimation problem using graph cuts. The result of the ML estimation depends on what statistical model we use. Under the Gaussian model, it is obtained by finding the means of the region features. It is shown that other statistical models can also fit in our framework. The main contributions of this work include: 1) a novel probabilistic model and an iterative optimization scheme for image segmentation, and 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. Our algorithm can cluster relevant regions in an image well, with the segmentation boundaries matching the region edges. Extensive experiments show that our algorithm can obtain results highly consistent with human perception. The qualitative and quantitative comparisons demonstrate that our algorithm outperforms six other state-of-the-art image segmentation algorithms.

II. A NEW PROBABILISTIC MODEL

In this section, we first introduce the features used to describe the properties of each pixel, and then present the new probabilistic model. For a given image P, the features of every pixel p are expressed by a four-dimensional vector

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

where (p), (p) and (p) are the components of p in the L*a*b* color space, and (p)denotes the texture feature of p. Several classical texture descriptors have been developed. In this paper, the texture contrast defined in [13] (scaled from [0; 1] to [0; 255]) is chosen as the texture descriptor. Fig. 1 shows an example of the features. The task of image segmentation is to group the pixels of an image into relevant regions. If we formulate it as a labeling problem, the objective is then to find a label configuration



f = f_p where is the label of pixel p denoting which region this pixel is grouped into. Generally speaking, a "good" segmentation means that the pixels within a region i should share homogeneous features represented by a vector Φ(i) that does not change rapidly except on the region boundaries. The introduction of Φ(i) allows the description of a region, with which high level knowledge or learned information can be incorporated into the segmentation. Suppose that we have k possible region labels.[2]

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

A four-dimensional vector is used to describe the properties of label (region) i, where the four components of Φ(i)have the similar meanings to those of the corresponding four components of I(p) and will be derived in Section II-B. let Φ={ Φ(i) }g be the union of the region features. If P and Φ(i) are known, the segmentation is to find an optimal label configuration b f, which maximizes the posterior possibility of the label configuration[3]-[5]..

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

where Φ 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 Φ that is precise enough. Our strategy here is to refine Φ according to the current label configuration found by (3). Thus, we propose to use an iterative method to solve the segmentation problem. Suppose that Φ^n and f^n are the estimation results in the nth iteration. Then the iterative formulas for optimization are defined as

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

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

This iterative optimization is preferred because (4) can be solved by the MAP estimation, and (5) by the ML estimation. Based on this framework, next we will explain how the MAP and ML estimations are implemented.

A. MAP Estimation of f from Φ:

Given an image P and the potential region features Φ, we infer f by the Bayesian law, i.e., Pr(f/ Φ,P) can be obtained by

$$\mathbf{Pr}(f|\Phi, P) = \frac{\mathbf{Pr}(\Phi, P|f)\mathbf{Pr}(f)}{\mathbf{Pr}(\Phi, P)} \propto \mathbf{Pr}(\Phi, P|f)\mathbf{Pr}(f), \quad (6)$$

which is a MAP estimation problem and can be modeled using MRFs. Assuming that the observation of the image follows an independent identical distribution (i.i.d.), we define Pr(Φ,P/f) as

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

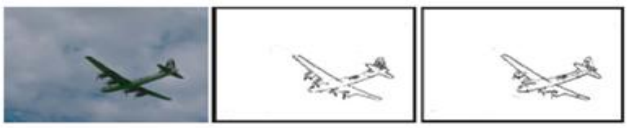


Figure 2: An example of the brightness contrast. (a) The original image. (b)
The brightness contrast in the horizontal direction. (c) The brightness contrast in the vertical direction

where D(p, fp, Φ) is the data penalty function which imposes the penalty of a pixel p with a label fp for given Φ The data penalty function is defined as

$$D(p, f_p, \Phi) = ||\mathbf{I}(p) - \boldsymbol{\phi}(f_p)||^2$$
$$= (I_L(p) - \overline{I}_L(f_p))^2 + (I_a(p) - \overline{I}_a(f_p))^2 + (I_b(p) - \overline{I}_b(f_p))^2 + (I_t(p) - \overline{I}_t(f_p))^2. \quad (8)$$

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

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

where N(p) is the neighborhood of pixel p$V_{pq}(f_p, f_q)$ called the smoothness penalty function, is a clique potential function, which describes the prior probability of a particular label configuration with the elements of the clique (p, q). We define the smoothness penalty function as follows using a generalized Potts model[6]-[8].

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

(10)

where Δ(p,q) = $I_L(P) - I_L(Q)$ called brightness contrast, denotes how different the brightness of p and q are, c > 0 is a smoothness factor, σ > 0 is used to control the contribution of Δ(p,q) to the penalty, and T(.) is 1 if its argument is true and 0 otherwise. From our experiments, we found that σ =2(Δ(p,q)) is a good choice, where <.> denotes the expectation of all the pairs of neighbors in an image. $V_{pq}(f_p, f_q)$depicts two kinds of constraints[11]. The first enforces the spatial smoothness; if two neighboring pixels are labeled differently, a penalty is imposed. The second considers a possible edge between p and q; if two neighboring pixels cause a larger Δ, then they have greater likelihood to be partitioned into two regions. Figure. 2 is an example of the brightness contrast. In our algorithm, the boundaries of the segmentation result are pulled to match the darker pixels in Figure. 2(b) and (c), which are more likely to be edge pixels. From (6), (7), and (9), we have

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

Taking the logarithm of (11), we have the following energy function[13].

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

It includes two parts: the data term

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

and the smoothness term

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

From (12), we see that maximizing Pr(f/ Φ,P) is equivalent to minimizing the Markov energy E(f, Φ) for a given Φ.[14]. In this paper, we use a graph cut algorithm to solve this minimization problem, which is described in Section III.

B. *ML Estimation of Φ from f:*

If the label configuration f is given, the optimal Φ should maximize Pr(f/ Φ,P) ,or minimize E(f, Φ) equivalently. Thus we have

$$\nabla_\Phi \log \mathbf{Pr}(f|\Phi, P) = \mathbf{0}, \quad (15)$$

Or

$$\nabla_\Phi E(f, \Phi) = \mathbf{0}, \quad (16)$$

Where Δ_Φ denotes the gradient operator. Since $V_{pq}(f_p, f_q)$ is independent of Φ, we obtain

$$\nabla_\Phi \sum_{p\in P} D(p, f_p, \Phi) = \mathbf{0}, \quad (17)$$

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

$$\sum_{p\in P} D(p, f_p, \Phi) = \sum_i \sum_{f_p=i} ||\mathbf{I}(p) - \boldsymbol{\phi}(i)||^2. \quad (18)$$

Therefore, (17) can be written as

$$\frac{\partial}{\partial\boldsymbol{\phi}(i)} \sum_{f_p=i} ||\mathbf{I}(p) - \boldsymbol{\phi}(i)||^2 = \mathbf{0}, \quad (19)$$

From (19), we obtain the ML estimation $\Phi = \Phi(i)$,where

$$\boldsymbol{\phi}(i) = \frac{1}{num_i}\sum_{f_p=i}\mathbf{I}(p), \quad (20)$$

Note that when the label configuration $f=\{f_p/p\}$ is unknown, finding the solution of (17) is carried out by clustering the pixels into groups. In this case, the ML estimation is achieved by the K-means algorithm [12], which serves as the initialization in our algorithm described in Section III.

C. *Non-Gaussian Modeling:*

The definition of $D(p, f_p, \Phi)$ in (8) uses the Gaussian model to describe a uniform region. Some other distributions in the modeling of natural images, such as the exponential family distributions [17], [18], can also be used in our framework. Let us take another popular model, the Laplace model [19], as an example. To replace the Gaussian model with the Laplace model, we modify (7) as

$$\mathbf{Pr}(\Phi, P|f) \propto \prod_{p\in P}\exp(-D'(p, f_p, \Phi)), \quad (21)$$

where the data penalty is defined as

$$D'(p, f_p, \Phi) = |\mathbf{I}(p) - \boldsymbol{\phi}(f_p)|. \quad (22)$$

With this data penalty, the MAP estimation is the same as when the Gaussian model is used. However, the ML estimation result is different from (20) and becomes

$$\boldsymbol{\phi}(i) = \textbf{Median}\{\mathbf{I}(p) | f_p = i\}, \quad (23)$$

where Median {.} denotes the median of the elements in a set [15]. In addition to the above parametric models, we can also use non-parametric distributions to describe the region features. Similar to the parametric models, the data penalty functions are defined as the negative logarithm of different likelihood functions in different non-parametric models (e.g., a histogram clustering model is used. In summary, different statistical models lead to different definitions of the data penalty. Given different data penalties, the MAP estimations are the same, but the ML estimation results depend on the used models. In the rest of this paper, we only consider the Gaussian model[16]-[19].

III. THE PROPOSED ALGORITHM

We first give the description of the algorithm for image segmentation, and then prove its convergence.

A. *Algorithm Description:*

With $E(f; \Phi)$ defined in (12), the estimations of b f and b Φ in (4) and (5) are now transformed to

$$f^{n+1} = \arg\min_{f} E(f, \Phi^n) \quad (24)$$

$$\Phi^{n+1} = \arg\min_{\Phi} E(f^{n+1}, \Phi) \quad (25)$$

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

Algorithm: Image segmentation:

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 Φ.
- **Step 3:** Iterative optimization.
  - **3.1:** MAP estimation — Estimate the label configuration f based on current Φ using the graph cut algorithm [36].
  - **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 Φ based on current f with (20).
- **Step 4:** If Φ 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 of the image.

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. For example, the upper-left and the lower-right regions are both labeled by 1. After step 3.2, each of the connected regions has a unique label. The MAP estimation is an NP-hard problem proposed to obtain an approximate solution via finding the minimum cuts in a graph model. Minimum cuts can be obtained by computing the maximum flow between the terminals of the graph. In [16], an efficient max-flow algorithm is given for solving the binary labeling problem. In addition, an algorithm, called α expansion with the max-flow algorithm embedded, is presented to carry out multiple labeling iteratively. In our algorithm, the α expansion algorithm is used to perform step 3.1. Besides the graph cuts, other techniques such as belief propagation can also be used to solve the MAP-MRF problem. One remarkable property of our algorithm is the ability to adjust the region number automatically during the iterative optimization with the relabeling step embedded into the MAP and ML estimations. Another property of our algorithm is that it is insensitive to the value of K in the initialization step with the K-means algorithm.

Now we analyze the computational complexity of the algorithm. In step 2, the K-means algorithm takes $O(NdKTk)$ time [12], where N is the number of pixels in an image, d is the number of features used to represent a pixel/region, K is the number of clusters, and Tk is the number of iterations. In our application, $d = 4$, K is set to 10, and Tk is set to 100. Both step 3.2 and step 3.3 take $O(N)$ time. In step 3, the main computational burden is the use of the graph cut algorithm (the α expansion) in step 3.1. The max-flow algorithm is linear in practice. The α expansion algorithm takes $O(NCnT\alpha n)$ time to carry out the MAP estimation during the n-th execution of step 3.1, where Cn is the number of label candidates and $T\alpha n$ is the number of iterations inside the α expansion. Let T be the number of executions of step 3.1. Then the computational complexity of our algorithm is $O(NdKTk)$+$O(NPT$ i=1 $CnT\alpha n)$. In general, Cn ranges from 1 to 50, $T\alpha n$ is less than 5, and T is less than 10.[20]

B. *Algorithm Convergence:*

We prove that the proposed algorithm is convergent in this section. Suppose that after the nth iteration, the energy is En, the configuration is fn, and the union of region features is Φn. The MAP estimation is to estimate the configuration fn+1 by minimizing the energy. Therefore, after the MAP estimation step of the (n+1)th iteration, the energy E^{n+1}_{MAP} decreases or keeps unchanged, i.e.,

$$E^{n+1}_{\text{MAP}} \le E^{n} \quad (26)$$

Suppose that the configuration is $f^{n+1}_{\text{relabeling}}$ after the relabeling step. This step only changes the labels of some regions but not their features, i.e., for each pixel p,

$$\boldsymbol{\phi}(f_p^{n+1}) = \boldsymbol{\phi}(f^{n+1}_{\text{prelabeling}}) \quad (27)$$

Therefore, from (8) and (13), the relabeling step does not change the data term. On the other hand, after the relabeling, for two neighboring pixels p and q.

$$T(f_p^{n+1} \neq f_q^{n+1}) = T(f^{n+1}_{\text{prelabeling}} \neq f^{n+1}_{\text{qrelabeling}}) \quad (28)$$

which implies that the relabeling step does not change the smoothness term either (see (10) and (14)). Thus, after the relabeling step, the energy keeps unchanged, i.e.,

$$E^{n+1}_{\text{relabeling}} = E^{n+1}_{\text{MAP}}. \tag{29}$$

Furthermore, since the ML estimation does not change the smoothness term but may reduce the data term or keeps it unchanged, we have

$$E^{n+1} \leq E^{n+1}_{\text{relabeling}}. \tag{30}$$

So the energy keeps monotonically non-increasing during the iterations, i.e.,

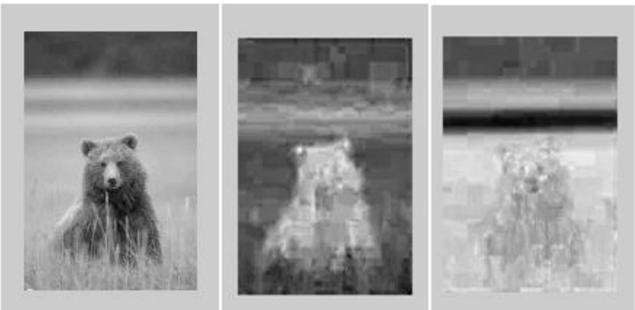
$$E^{n+1} \leq E^{n}, \tag{31}$$

which completes the proof of the convergence of our algorithm.

IV. MATLAB RESULTS

We test the proposed algorithm by using Matlab, We classify part of the image in the Matlab" a dog" and show the segmentation results obtained by the algorithm in Figure 3 All the boundaries of the small regions with the numbers of pixels less than 100 are removed. From these example, we have the following observations.

(a)



(b)

Figure 3 :a), (b) Segmentation results on the " dog"

Quantitative comparisons are also important for objectively evaluating the performance of the algorithm. There have been several measures proposed for this purpose. Region differencing and boundary matching are two of them. Region differencing measures the extent to which one segmentation can be viewed as a refinement of the other. Boundary matching measures the average displacement error of boundary pixels between the results obtained by an algorithm and the results obtained from human subjects. However, these two measures are not good enough for segmentation evaluation. For example, a segmentation result with each pixel being one region obtains the best score using these two measures. A strongly over-segmented result, which does not make sense to our visual perception, may be ranked good.

V.CONCLUSION

In this paper, we have developed a Extensive Technique for image segmentation algorithm. Our algorithm is formulated as a labeling problem using a probability maximization model. An iterative optimization technique combining the MAP and ML estimations is employed in our framework. Under the Gaussian model, the MAP estimation problem is solved using graph cuts and the ML estimation is obtained by finding the means of the region features. The qualitative and results demonstrate that our algorithm out performs than others. Our future work includes the extension of the proposed model to video segmentation with the combination of motion information technique , and the utilization of the model for specific object extraction by designing more complex features (such as shapes) to describe the objects.

References

[1] Shifeng Chen, Liangliang Cao, Yueming Wang, Jianzhuang Liu, and Xiaoou Tang" Image Segmentation by MAP-ML Estimations" IEEE Trans on Image processing Vol No 19, pp 2254-2264 Sept 201.

[2] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *Int'l J. Computer Vision*, vol. 1, no. 4, pp. 321–331, 1988.

[3] L. Cohen *et al.*, "On active contour models and balloons," *CVGIP: Image Understanding*, vol. 53, no. 2, pp. 211–218, 1991.

[4] C. Xu and J. Prince, "Snakes, shapes, and gradient vector flow," *IEEE Trans. Image Processing*, vol. 7, no. 3, pp. 359–369, 1998.

[5] E. Mortensen and W. Barrett, "Interactive segmentation with intelligent scissors," *Graphical Models and Image Processing*, vol. 60, no. 5, pp. 349–384, 1998.

[6] T. McInerney and D. Terzopoulos, "Deformable models in medical image analysis: A survey," *Medical Image Analysis*, vol. 1, no. 2, pp. 91–108, 1996.

[7] D. Cremers, M. Rousson, and R. Deriche, "A review of statistical approaches to level set segmentation: Integrating color, texture, motion and shape," *International Journal of Computer Vision*, vol. 72, no. 2, pp. 195–215, 2007.

[8] X. Fan, P. Bazin, and J. Prince, "A multi-compartment segmentation framework with

homeomorphic level sets," in *IEEE Proc. Computer Vision and Pattern Recognition*, 2008.

[9] J. Olszewska, C. De Vleeschouwer, and B. Macq, "Speeded up gradient vector flow b-spline active contours for robust and real-time tracking," in *IEEE Proc. Int'l Conf. Acoustics, Speech and Signal Processing*, 2007.

[10] S. Beucher and F. Meyer, "The morphological approach to segmentation: The watershed transformation," *Mathematical Morphology in Image Processing*, vol. 34, pp. 433–481, 1993.

[11] L. Vincent and P. Soille, "Watersheds in digital spaces: An efficient algorithm based on immersion simulations," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, no. 6, pp. 583–598, 1991.

[12] R. Duda, P. Hart, and D. Stork, *Pattern Classification*, 2nd ed. Wiley- Interscience, 2001.

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

[14] D. Comaniciu and P. Meer, "Mean shift analysis and applications," in *IEEE Proc. Int'l Conf. Computer Vision*, vol. 2, 1999, pp. 1197–1203.

[15] D. Comaniciu and P. Meer, "Mean shift: a robust approach toward feature space analysis," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002.

[16] A. Y. Yang, J. Wright, S. S. Sastry, and Y. Ma, "Unsupervised segmentation of natural images via lossy data compression," EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2006-195, Dec 2006. [Online]. Available: http://www.eecs.berkeley.edu/Pubs/TechRpts/2006/EECS-2006-195.html

[17] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[18] S. X. Yu and J. Shi, "Multiclass spectral clustering," in *IEEE Proc. Int'l Conf. Computer Vision*, vol. 1, 2003, pp. 313–319.

[19] Z. Wu and R. Leahy, "An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 15, no. 11, pp. 1101–1113, 1993.

[20] P. Felzenszwalb and D. Huttenlocher, "Efficient graph-based image segmentation," *Int'l J. Computer Vision*, vol. 59, no. 2, pp. 167–181, 2004.

[21] Z. Tu and S.-C. Zhu, "Image segmentation by data-driven Markov chain Monte Carlo," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 657–673, 2002.