Graph cut-based multiple active contours without initial contours and seed points

Jong-Sung Kim[O], Ki-Sang Hong
Division of Electrical and Computer Engineering, POSTECH
Kimjs@postech.ac.kr, Hongks@postech.ac.kr

Abstract

This paper presents a new graph cut-based multiple active contour algorithm to detect optimal boundaries and regions in images without initial contours. The task of multiple active contours is framed as a partitioning problem by assuming that image data are generated from a finite mixture model with unknown number of components. Then, the partitioning problem is solved within a divisive graph cut framework where multi-way minimum cuts for multiple contours are efficiently computed in a top-down way through a swap move of binary labels. A split move is integrated into that framework to estimate the model parameters associated with regions without the use of initial contours and seed points. The number of regions is also estimated as a part of the algorithm. Experimental results of boundary and region detection of natural images are presented to demonstrate the effectiveness of the proposed algorithm.

1. Introduction

Most of the works for boundary and region detection from images have been done by using active contours [4, 5], which detect region boundaries by evolving initial contours or seed points close to true ones toward the closest edges or toward the boundary between two homogeneous regions. However, such initial contours are far to seek without human assistance. Clearly, it limits the use of active contours, especially for detecting the complex boundary of a number of regions. Therefore, this work focused on how to avoid the use of initial contours or seed points.

The advantages of interactive graph cuts [1] and active contours have been combined to detect optimal boundaries in images [2]. User-specified seed points or contours have been used to define source and sink nodes of graph and used to compute the regional property of object and background [1, 2]. It has been extended to the boundary detection of multiple regions using the interactive multi-way graph cuts [1]. Although the graph cut method dramatically improved the performance of active contours, the problem associated with initial contours or seed points remained unsolved.

The aim of this work is to investigate a new graph cut-based multiple active contour (MAC) algorithm without initial contours and seed points. For the aim, a new graph cut framework is introduced to avoid the use of the interactive multi-way graph cuts [1-3], which is inadequate to this work, since it is hard to set the sink and source nodes of multi-way graph without human assistance and prior knowledge, especially the number of labels. In the new framework, the task of MAC is framed as divisive partitioning (clustering) which parts given data into a hierarchy of groups, each with different properties, in a top-down (divisive) way. Other partitioning and clustering techniques have been studied for automatic image segmentation, e.g., the normalized cuts [9] and the mean shift algorithm [6], but their performances of boundary and region detection were not satisfactory. Fig. 1 illustrates the proposed framework with an example, which shows that the source image is divisively partitioned into similar color regions with contours (partitioning cuts) representing region boundaries.

The MAC model of this work assumes that image data are generated from a finite mixture model with unknown number of components. Then, the MAC model is formulated in terms of energy minimization and solved within the divisive graph cut framework as follows: The image is partitioned into two regions by computing a minimum cut of the energy with a swap move of binary labels [3], and then the same procedure is recursively applied to each region to obtain new regions until some stopping conditions are met. A split move is integrated with the swap move to set the initial values of the model parameters. Usual split moves based on random sampling, e.g., see [7], are computationally too expensive. Instead, the split move of this work employs a logistic model to split regions and the logistic model is learned unsupervisedly from the first and second moments of image data. The effectiveness of the proposed algorithm is demonstrated with the boundary and region detection results of some natural images from the Berkeley image segmentation database [8]. The results of the proposed algorithm are compared with those of the normalized cut algorithm and the mean-shift algorithm.

This paper is organized as follows. Section 1 describes the proposed multiple active model. Section 2 introduces the divisive graph cuts framework and explains the swap move and the split move within that framework. Sections 4 and 4 present experimental results and conclusions.

2. The multiple active contour model

The aim of the MAC model is to partition the overall region R of a given image into K disjoint regions $\{R_1,\ldots,R_K\}$, with K contours $\{C_1,\ldots,C_K\}$ corresponding to the boundary of each of that regions. The number K is unknown in general cases. For this aim, the image is modelled with a finite mixture model with K components

Fig. 1 An illustration of the divisive graph cut framework.

and each region is assigned to one of the K components of the mixture model. Low-level image features, e.g., intensity, color, texture, and motion, can be used for that image model. In this work, RGB color is selected as image feature, since it is the most common and important feature among those and also widely used in other active contour models [4, 5]. The image is an array $\mathbf{z}=\{\mathbf{z}_1,\ldots,\mathbf{z}_p,\ldots,\mathbf{z}_{|R|}\}$ of image data in RGB color space, where p is the index of data and $|R|$ the number of data in the region R. Then, the distribution of that image data is a mixture density model with unknown K components:

$$f\left(\mathbf{z}_p \mid \Theta\right)=\sum_k \alpha_k f_k\left(\mathbf{z}_p \mid \theta_k\right), \tag{1}$$

where $\Theta=\{(\alpha_1,\theta_1),\ldots,(\alpha_K,\theta_K)\}$ is an unknown set consisting of mixing weights a_k and model parameters θ_k given a parametric density model $f_k(\cdot\,|\theta_k)$ for the k-th component density model. However, the mixture model does not describe from which component each datum $\mathbf{z}_p$ is generated. To indicate the assignment of each datum, a label vector $c=\{c_p|p\in R\}$ with $c_p \in \{l_1,\ldots,l_K\}$ and $c_p = l_k$ if $p\in R_k$ is introduced. Given the value of c_p, each image datum $\mathbf{z}_p$ is generated from one of the components of the mixture distribution. Therefore, the mixture model can be written as a product

$$f\left(\mathbf{z}_p \mid \Theta, c_p\right)=\prod_k\left(\alpha_k f_k\left(\mathbf{z}_p \mid \theta_k\right)\right)^{\delta\left[c_p=l_k\right]}, \tag{2}$$

where the Kronecker delta $\delta[c_p= l_k]=1$ if $c_p= l_k$, and $\delta [c_p= l_k]=0$ otherwise. The above label vector c implicitly represents the K contours by indicating the inner region of each contour with the label value as the level set functions do in the level set method. The label vector c work as a latent variable in the parameter estimation of mixture models via the expectation maximization (EM) algorithm, of which details will be explained later.

The MAC model is formulated in terms of energy minimization as follows. A discrete energy E is defined over the model parameters Θ and the label vector c. Then, the MAC are computed by alternatively minimizing the energy E with respect to Θ and c. By combining the regional and boundary properties of image data, the energy function E is defined by

$$E(\Theta, c)=\lambda \cdot A(\Theta, c)+B(c), \tag{3}$$

where A and B are the regional and boundary energy terms, respectively, and a positive constant λ balances the importance between the two energy terms. The regional energy term A is defined by

$$A(\Theta, c)=\sum_{p\in R} A_p, \tag{4}$$

where

$$A_p=-\ln f\left(\mathbf{z}_p \mid \Theta, c_p\right). \tag{5}$$

Each component of the mixture model can be modelled with a normal density function, which is common in finite mixture models [7]. In addition, the presence of outliers can be modelled by incorporating robust statistical techniques into the model, which is important for practical applications, since statistical outliers may be encountered in image data. Then, the component model $f_k(\cdot\,|\theta_k)$ is defined for all k to be an elliptically symmetric density function with mean vector $\mathbf{m}_k$ and covariance matrix S_k as

$$f_k\left(\mathbf{z}_p \mid \mathbf{m}_k, S_k\right) \propto \frac{1}{\det S_k^{1/2}} \exp\left(-\rho\left(r_{p,k}\right)\right), \tag{6}$$

where $\rho(\cdot)$ is the Huber influence function and the square of the residual $r_{p,k}$ is defined by

$$r_{p,k}^2\left(\mathbf{z}_p, \mathbf{m}_k, S_k\right)=\frac{1}{2}\left(\mathbf{z}_p-\mathbf{m}_k\right)^T S_k^{-1}\left(\mathbf{z}_p-\mathbf{m}_k\right). \tag{7}$$

The MAC should be localized along with image edges as in other active contour models [4]. In addition, the computation of geodesics or minimal length curves is also demanded to regularize the shape of the contours [2, 5]. The above two boundary constraints for the MAC will be integrated into the boundary energy term B, which is defined by

$$B(c)=\sum_{(p,q)\in N} B_{p,q}\cdot\delta\left[c_p \neq c_q\right], \tag{8}$$

where N is the set of is the set of all pairs of neighboring pixels, which is varied with the size of neighborhood system, and $\delta[c_p\neq l_k]=1$ if $c_p\neq l_k$ and $\delta[c_p\neq l_k]=0$ otherwise, which represents the Potts interaction model to impose a penalty for discontinuity between p and q of similar feature. This interaction model encourages the labelling c consisting of several regions by enforcing pixels of the same region to have equal labels.

The function $B_{p,q}$ is defined as a product of two functions

$$B_{p,q}=G_{p,q}\cdot W_{p,q}, \tag{9}$$

where two functions $G_{p,q}$ and $W_{p,q}$ specifies the above two boundary properties of the MAC, respectively. The first function $G_{p,q}$ is defined by

$$G_{p,q}=\nu\cdot\exp\left(-\frac{1}{2}d_Q\left(\mathbf{z}_{p,h}, \mathbf{z}_{q,h}\right)^2\right)+1-\nu, \tag{10}$$

where $\nu=[0,1]$ is a constant, $d_Q(\cdot)$ denotes the distance normalized by a matrix Q, and $\mathbf{z}_p$ the feature vector filtered with a Gaussian kernel at the scale of h pixels. If the constant ν of the model is set to zero, the influence of image edges can be nullified as in the active contour model without edges [5], which is advantageous when edges are unavailable or unreliable. As shown in Fig. 2(a), contours

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

Fig. 2 Example contours obtained with the proposed model. (a) v=0. (b) v=1, h=0. (c) v=0.9, h=0. (d) v=1, h=1.

can be obtained with v=0, while it may not account image boundaries. The influence of image boundaries is maximized when v=1. However, superfluous boundaries and regions may occur at some pixels of which the boundary strength is too high (see Fig. 2(b)). This problem can be avoided by setting the value of ν below one as in Fig. 2(c). Additionally, a smoothing procedure of image data can be performed with a Gaussian filter of scale σ a priori. This smoothing procedure is also effective against superfluous boundaries and regions, which is illustrated in Fig. 2(d). The matrix Q has been estimated as a covariance matrix with the value of $\mathbf{z}_{p,h}$ - $\mathbf{z}_{q,h}$ for a given scale factor σ and all pairs of p and q in N. This function penalizes a lot discontinuity between p and q of which the image data are similar to each other. The penalty function $G_{p,q}$ enforces the solution c to partition the given image along with image edges.

In a regular and equal spacing grid graph with a neighborhood system, the second penalty function $W_{p,q}$ is defined by

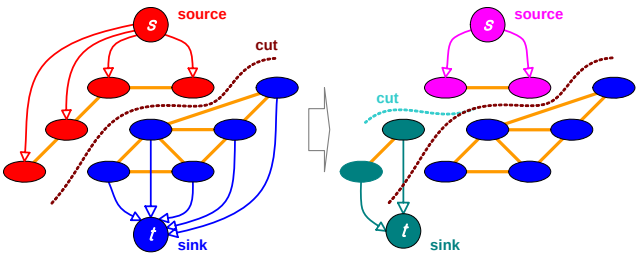
$$W_{p,q} = \frac{\delta^2 \cdot \Delta\phi_{p,q}}{2 \cdot d_I(p,q)}, \qquad (11)$$

where δ is the cell size, $\Delta\phi_{p,q}$ the angular difference to the nearest family of edge-lines given the grid, $d_I(p,q)$ the distance between p and q and I the identity matrix. The cell size δ is usually set to one and the angular difference $\Delta\phi_{p,q} \approx 2\pi/m$, where m is the size of neighborhood system, e.g., m=16 in case of 16-neighborhood system. The function $W_{p,q}$ computes the Euclidean cut metric between p and q, which approximates the length of curves based on the Cauchy-Crofton formula from integral geometry [2].

3. The divisive graph cut framework

The graph cut algorithm [3] can find an optimal labelling c of the energy E following some procedure to estimate the value of model parameters Θ. Many estimation techniques, such as the EM algorithm, can be applied to the parameter estimation if initial values are available. However, it is difficult to set initial values without prior knowledge, such as initial contours or seed points. In addition, the image is modelled with the mixture of K normal distribution. The component number K can be given a prior in some cases, e.g., in the case of interactive segmentation [1, 5], but it is unknown in general cases.

The divisive graph cut framework overcomes the above difficulties of the MAC model. The basic idea is to solve the multiple labelling problem in the MAC model as a binary labelling problem. The optimal solution to the binary labeling problem can be found by using any min-cut/max-flow algorithms solving s-t minimum cuts on graphs as illustrated in Fig. 3. In a similar top-down way, the K components of the mixture model are learned by starting with two components.



s-t minimum cut new s-t minimum cut

Fig. 3 Divisive graph cuts. A s-t minimum cut algorithm is recursively applied to subgraphs to obtain new s-t minimum cuts.

Let's denote R_0 a current region and N_0 the set of all pairs of neighboring pixels in R_0, where $R_0 \subset R$ and $N_0 \subset N$. Then, the task of the divisive graph cut algorithm is to partition the current region R_0 into two new subregions R_1 and R_2, s.t. $R_1 \cup R_2 = R_0$ and $R_1 \cap R_2 = \emptyset$, according to a simplified MAC model, which is now defined over R_0 and N_0 with respect to a label vector c_0 and model parameters Θ_0. The simplified energy E_0 at the current stage is

$$E_0(\Theta_0, c_0) = \lambda \cdot \sum_{p \in R_0} A_p + \sum_{(p,q) \in N_0} B_{p,q} \cdot \delta[c_p \neq c_q], \qquad (12)$$

where $\Theta_0 = \{(\alpha_1,\theta_1),(\alpha_2,\theta_2)\}$ and the label vector c_0 is binary-valued, i.e. $c_0 = \{c_p | p \in R_0\}$ with $c_p = l_1$ if $p \in R_1$ and $c_p = l_2$ if $p \in R_2$. Given an initial estimate of the model parameters Θ_0, an optimal labeling c_0^* can be obtained by minimizing E_0 over all labellings within one swap move of c_0 as in [3] where an approximate solution to the multiple labelling problem is iteratively obtained over all labellings with the swap move of each pair of possible labels.

Let $\mathcal{G}_0 = (\mathcal{V}_0, \mathcal{E}_0)$ be a weighted graph with two distinguished terminals, source s and sink t, where $\mathcal{V}_0$ is a set of nodes and $\mathcal{E}_0$ a set of edges. Then, the optimal labelling c_0 corresponds to a s-t minimum cut on a graph $\mathcal{G}_0 = (\mathcal{V}_0, \mathcal{E}_0)$, where the set of nodes $\mathcal{V}_0$ is the sum of the set of pixels at the current region R_0 and terminal nodes s and t corresponding to two labels l_1 and l_2:

$$\mathcal{V}_0 = R_0 \bigcup \{s, t\}. \qquad (13)$$

Two types of edges, n-links and t-links, make up the set of edges $\mathcal{E}_0$, where each pixel p in R_0 has two t-links, $\{p, s\}$

and $\{p, t\}$, and each pair of neighboring pixels $\{p, q\}$ in N_0 is connected by a n-link, which is also denoted by $\{p, q\}$. Therefore, the set of edges $\mathcal{E}_0$ is

$$\mathcal{E}_0 = N_0 \bigcup \{\{p,\ s\},\ \{p,\ t\}\}. \tag{14}$$

To completely define the graph $\mathcal{G}_0$ for the s-t minimum cut computation, the weights of edges in $\mathcal{E}_0$ are listed in the following table. Then, any min-cut/max-flow algorithms can be applied to compute a minimum cost cut on two terminal graphs [3].

edge	weight	for
$\{p, s\}$	$\lambda \cdot \left(-\ln \alpha_1 + \frac{1}{2} \ln \det S_1 + \rho\left(r_{p,1}\right) \right)$	$p \in R_1$
$\{p, t\}$	$\lambda \cdot \left(-\ln \alpha_2 + \frac{1}{2} \ln \det S_2 + \rho\left(r_{p,2}\right) \right)$	$p \in R_2$
$\{p, q\}$	$\left(\nu \cdot \exp\left(-\frac{1}{2} d_Q\left(\mathbf{z}_{p,h}, \mathbf{z}_{q,h}\right)^2 + 1 - \nu \right) \right) \times \frac{\delta \cdot \Delta\phi_{p,q}}{2 \cdot d_I(p,q)}$	$\{p, q\} \in N_0$

A parameter estimation step follows the above labelling step to obtain the optimal model parameters Θ_0^* using the obtained optimal labelling c_0^*, where the two successive steps, i.e. the estimation and labelling steps, composes one iteration of swap move of the proposed algorithm, which is indexed by n to present clearly. The parameter estimation is performed using the EM algorithm as follows. First, the current parameter values are computed as

$$\alpha_i^{(n)} = \frac{\sum_{p \in R_0} \delta\left[c_p^* = l_i\right]}{\sum_{j \in \{1,\ 2\}} \sum_{p \in R_0} \delta\left[c_p^* = l_j\right]} \tag{15}$$

$$\mathbf{m}_i^{(n)} = \frac{\sum_{p \in R_0} \mathbf{z}_p \cdot \delta\left[c_p^* = l_i\right]}{\sum_{p \in R_0} \delta\left[c_p^* = l_i\right]} \tag{16}$$

$$S_i^{(n)} = \frac{\sum_{p \in R_0} \left(\mathbf{z}_p - \mathbf{m}_i^{(n)}\right)\left(\mathbf{z}_p - \mathbf{m}_i^{(n)}\right)^T \cdot \delta\left[c_p^* = l_i\right]}{\sum_{p \in R_0} \delta\left[c_p^* = l_i\right]} \tag{17}$$

for i=1, 2. The current parameter values are used as the initial estimates for the EM algorithm. Then, the initial estimates are iteratively updated using the following E-step and M-step:

E-Step:

$$\eta_{p,i}^{(n)} = \frac{\alpha_i^{(n)} f_i\left(\mathbf{z}_p \middle| \mathbf{m}_i^{(n)}, S_i^{(n)}\right)}{\sum_{j \in \{1,2\}} \alpha_j^{(n)} f_j\left(\mathbf{z}_p \middle| \mathbf{m}_j^{(n)}, S_j^{(n)}\right)}, \tag{18}$$

where $\eta_{p,i}{}^{(n)}$ is the posterior probability that each pixel p belongs to the region R_i.

$$w_{p,i}^{(n)} = \begin{cases} 1 & \text{if } \left|r_{p,i}^{(n)}\right| \le \varepsilon_i^{(n)} \\ \varepsilon_i^{(n)} / r_{p,i}^{(n)} & \text{o.w.} \end{cases}, \tag{19}$$

where $w_{p,i}{}^{(n)}$ is the weight assigned to each pixel p, which is inversely proportional to $r_{p,i}{}^{(n)}$. The tunning constant $\varepsilon_i{}^{(n)}$ is computed with $\varepsilon_i{}^{(n)} = \beta \cdot \text{median}_p |r_{p,i}{}^{(n)}|$, where β=1.4826 typically.

M-Step:

$$\alpha_i^* = \frac{\sum_{p \in R_0} \eta_{p,i}^{(n)}}{|R_0|} \tag{20}$$

$$\mathbf{m}_i^* = \frac{\sum_{p \in R_0} \mathbf{z}_p \cdot \eta_{p,i}^{(n)} w_{p,i}^{(n)}}{\sum_{p \in R_0} \eta_{p,i}^{(n)} w_{p,i}^{(n)}} \tag{21}$$

$$S_i^* = \frac{\sum_{p \in R_0} \left(\mathbf{z}_p - \mathbf{m}_i^*\right)\left(\mathbf{z}_p - \mathbf{m}_i^*\right)^T \cdot \eta_{p,i}^{(n)} w_{p,i}^{(n)}}{\sum_{p \in R_0} \eta_{p,i}^{(n)} w_{p,i}^{(n)}}, \tag{22}$$

where the product of two weighting factors $\eta_{p,i}{}^{(n)}\ w_{p,i}{}^{(n)}$ replaces the delta function $\delta[c_p{}^* = l_i]$ with a confidence measure how well label l_i fits pixel p given the observed data and the estimated parameters. In general, EM steps perform until convergence. Although the convergence is theoretically guaranteed, an additional process to set the initial value of $\Theta_0{}^{(0)}$ without the value of $c_0{}^{(0)}$ at the first step is inevitably needed. For the aim, a split move is introduced.

Let $\mathbf{m}_0$ and S_0 be the mean vector and the covariance matrix of the image data in R_0, respectively. The mixing weights α_1 and α_2 and mean vectors $\mathbf{m}_1$ and $\mathbf{m}_2$ should satisfy that $\alpha_1+\alpha_2=1$ and $\alpha_1\mathbf{m}_1+\ \alpha_2\mathbf{m}_2=\mathbf{m}_0$. If the initial value of component weights are set to $\alpha_1=\ \alpha_2\ =1/2$. the mean vector $\mathbf{m}_0$ becomes the average of two unknown vectors $\mathbf{m}_1$ and $\mathbf{m}_2$, i.e. $\mathbf{m}_0=(\mathbf{m}_1+\ \mathbf{m}_2)/2$. Let the covariance matrices S_1 and S_2 be $S_1=\ S_2=\ \sigma^{-2}I$, where the constant σ is the standard deviation of noise. Let's define a difference vector $\Delta\mathbf{m}$ between $\mathbf{m}_1$ and $\mathbf{m}_2$ as $\Delta\,\mathbf{m}=\mathbf{m}_1-\mathbf{m}_2$ and assume that the density models $f_1(\ \cdot\ |\theta_1{}^{(0)})$ and $f_2(\ \cdot\ |\theta_2{}^{(0)})$ are usual normal distributions, which implies that $w_{p,1}{}^{(0)}=w_{p,2}{}^{(0)}=1$ for all p in R_0. Then, the posterior probability $\eta_{p,1}{}^{(0)}$ (or $\eta_{p,2}{}^{(0)}$) in the E-step can be rewirtten as a logistic function

$$\eta_{p,1}^{(0)} = \frac{\exp\left(-r_{p,1}^2\right)}{\exp\left(-r_{p,1}^2\right) + \exp\left(-r_{p,2}^2\right)} = \frac{1}{1+\exp\left(-a_p\right)}, \tag{23}$$

where

$$a_p = \sigma^{-2} \cdot \Delta\mathbf{m}^T \left(z_p - \mathbf{m}_0\right), \tag{24}$$

and the other probability $\eta_{p,2}{}^{(0)}$ (or $\eta_{p,1}{}^{(0)}$) is simply calculated with $\eta_{p,2}{}^{(0)}=1-\ \eta_{p,1}{}^{(0)}$. Now, the first initial value $\Theta_0{}^{(0)}$ can be computed using the M-step with the value of $\eta_{p,1}{}^{(0)}$ and $\eta_{p,2}{}^{(0)}$ for all p in R_0. More easily, the value of $\Theta_0{}^{(0)}$ can be initialized using the value of the initial labelling $c_0{}^{(0)}$, because the label of each element $c_{p,0}{}^{(0)}$ of the vector $c_0{}^{(0)}$ can be initialized by

$$c_{p,0}^{(0)} = \begin{cases} l_1 & \text{if } a_p \ge 0 \\ l_2 & \text{if } a_p < 0 \end{cases}, \tag{25}$$

The sign of a_p can be estimated even though the value of σ and the magnitude of $\Delta\mathbf{m}$ are unknown**.** The value of $\mathbf{m}_0$ is easily computed on the original data. The direction of $\Delta\mathbf{m}$ has a high possibility of the coincidence with the direction of the eigenvector with the largest eigenvalue of the covariance matrix S_0 [7], the unknown vector $\Delta\mathbf{m}$ is replaced by that eigenvector.

The split move and the swap move described above can partition a current region into two subregions automatically. This is the basic procedure of the proposed MAC algorithm. This procedure is recursively applied to each

Fig. 4 Results of the first partition corresponding to K=2 using the split move and the swap move in the proposed MAC algorithm. (a) 0.7 sec. (b) 0.6 sec. (c) 2.2 sec. (d) 1.3 sec. (e) 0.7 sec.

Fig. 5 Final results of automatic boundary and region detection using the proposed MAC algorithm. (a) 3.7 sec. (b) 4.6 sec. (c) 7.0 sec. (d) 6.5 sec. (e) 10.0 sec.

subregion until the number of regions K reaches to a predefined value $K_{\max}$ or a new region of the size lager than $\gamma|R|$ is no more detected.

4. Experimental results

For experiments, some natural color images shown in Fig. 4 were obtained from the Berkeley image segmentation database [8]. To reduce the computational cost of the graph cut algorithm, all test images was resized to 240×160 pixels. All computations were carried out on a desktop PC with a 2.13-GHz Intel Core2 CPU. The graph cut and energy minimization libraries [3, 10] were used for algorithm implementation. The value of λ was chosen to give the best performance for each image. The other constant parameters were set with h=1, $K_{\max}$ =30 and γ=0.01 equally for all test images. The value of v was set with v=1 for the images except the one in Fig. 4(e), where v=0.8 was used to reduce the influence of some high-contrasted pixels. The 16-neighborhood system was used to construct a regular grid-fashioned, undirected graph for each image. The weight for each edge was set as described in the preceding section. The detection results of the proposed MAC algorithm was compared with those of the normalized cut (NC) algorithm [9] and the mean shift (MS) algorithm [6]. The NC and MS softwares provided by the authors of each work were used. The bandwidth parameters h_s and h_r of the MS algorithm were also chosen to give the best performance for each image. The number of regions for each image was determined automatically in the MAC and MS algorithms, but it was specified a priori in the NC algorithm.

Fig. 4 shows the results of the first partition. Although the test images had multiple regions, they were partitioned successfully into two regions with contours localized correctly along with image edges as wanted in the

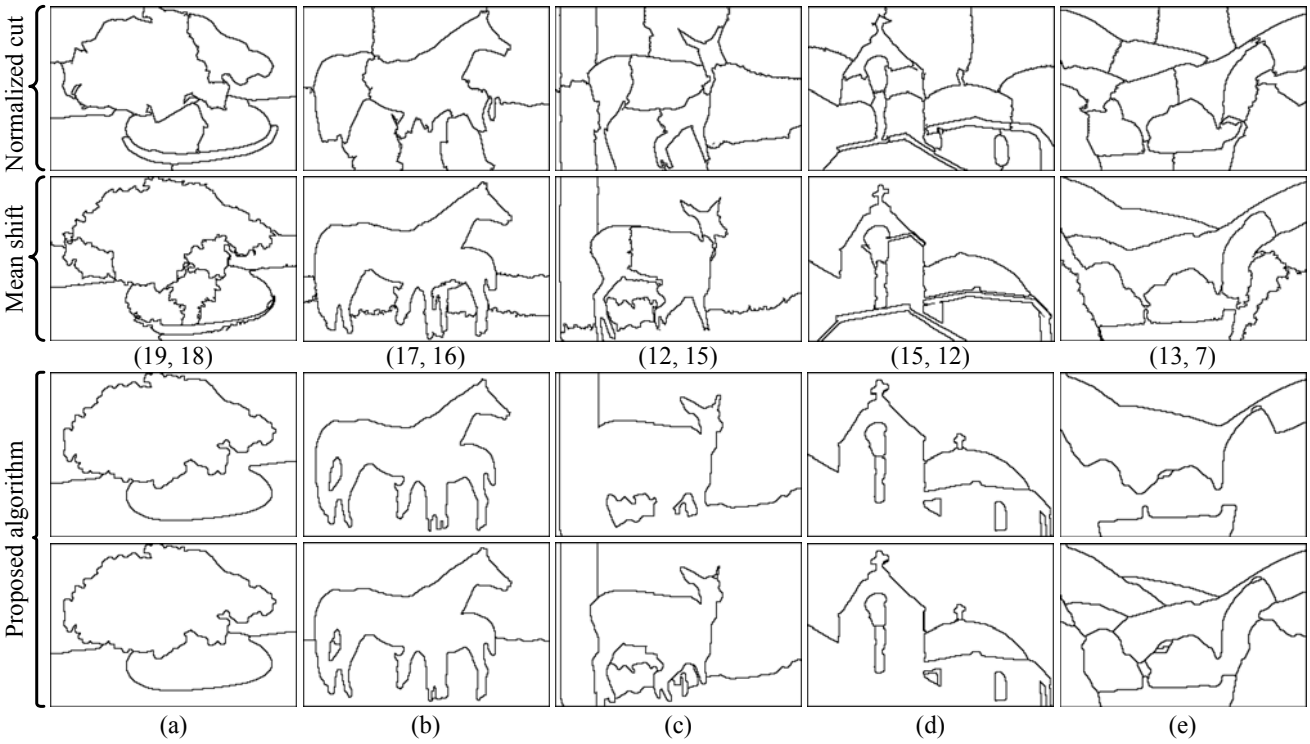


Fig. 6 Comparison of boundary and region detection of the proposed MAC algorithm with the normalized cut algorithm and the mean shift algorithm. (a) NC: 19.4 sec., MS: 2.5 sec. (b) NC: 19.5 sec., MS: 1.8 sec. (c) NC: 21.1 sec., MS: 1.0 sec. (d) NC: 23.3 sec., MS: 1.3 sec. (e) NC: 19.1 sec., MS: 2.8 sec.

proposed algorithm. The swap move computed a partition accurately, even though the split move gave a little inaccurate partition. The final results of boundary and region detection were in Fig. 5. The numbers of detected regions from each image were 4, 3, 5, 6, and 10. In Fig. 6, the boundary detection of the MAC was compared to the NC and MS. To show that the MAC made a sequence of meaningful outputs, an output obtained prior to the final one was displayed in the third row. In all cases, the MAC detected more smooth boundaries accurately than the NC and MS. The MAC was slower than the MS but faster than the NC.

5. Conclusion

A new graph cuts-based multiple active contour algorithm was presented. The use of initial contours and seed points was not required contrary to other algorithms for active contours or interactive segmentation. The proposed algorithm showed better performance in detecting boundaries than other state-of-the-art techniques. The proposed identified smooth boundaries as contours and also generated a sequence of meaningful outputs.

6. Reference

[1] Y. Boykov and G. Funka-Lea, “Graph cuts and efficient N-D image segmentation”, Int. J. Compt. Vision, Vol. 70, pp. 109-131, 2006

[2] Y. Boykov and V. Kolmogorov, “Computing geodesics and minimal surfaces via graph cuts”, In Proc. ICCV, 2003

[3] Y. Boykov, O. Veksler, and R. Zabih, “Fast approximate energy minimization via graph cuts”, IEEE Trans. Pattern Anal. and Mach. Intell., Vol. 23, pp. 1222-1239, 2001

[4] V. Caselles, R. Kimmel, and G. Sapiro, “Geodesic active contours”, Int. J. Compt. Vision, Vol. 21, pp. 61-79, 1997

[5] T. Chan and L. Vese, “Active contours without edges”, IEEE Trans. Image Process, Vol. 10, pp. 266-277, 2001

[6] D. Comaniciu and P. Meer, “Mean shift: A robust approach toward feature space approach”, IEEE Trans. Pattern Anal. and Mach. Intell., Vol. 24, pp. 603-619, 2002

[7] P. Dellaportas and I. Papageorgiou, “Multivariate mixtures of normals with unknown number of components”, Statis. Comput., Vol. 16(1), pp. 57-68, 2006

[8] D. Martin, C. Fowlkes, D. Tal, and J. Malik, “A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics”, In Proc. ICCV, pp. 416-423 2001

[9] J. Shi and J. Malik, “Normalized cuts and image segmentation”, IEEE Trans. Pattern Anal. and Mach. Intell., Vol. 22, pp. 888-905, 2000

[10] R. Szeliski, R. Zabih, R., D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother, “A comparative study of energy minimization methods for Markove random fields”, In Proc. ECCV, pp. 16-29, 2006