Unsupervised Grow-Cut: Cellular Automata-based Medical Image Segmentation

Payel Ghosh, Sameer K. Antani, L. Rodney Long, George R. Thoma
Lister Hill Center for Biomedical Communications
U. S. National Library of Medicine, National Institutes of Health
Bethesda, MD, USA
{ghoshp3, santani, rlong, gthoma}@mail.nih.gov

Abstract— **This paper presents a new cellular automata-based unsupervised image segmentation technique that is motivated by the interactive grow-cut algorithm. In contrast to the traditional method which requires user-interaction to identify classes, the unsupervised grow-cut algorithm (UGC) starts with a random number of seed points and automatically converges to a natural segmentation. This is useful when deriving classes from large image datasets for applications such as region-based image retrieval. The algorithm has been tested on a subset of thirty medical images derived from the ImageCLEFmed database and 300 natural images from the Berkeley dataset. The unsupervised grow-cut algorithm has been compared against the Mean Shift method and Normalized Cut method. The segmentation outcome of the UGC algorithm is comparable with the other two methods. The number of classes derived by the UGC is independent of the number of initial seed points. Incorporating cellular automata makes the computational complexity of the algorithm independent of the dimension of the image and feature space.**

I. INTRODUCTION

Segmentation is the process of deriving meaningful regions from images that are homogeneous with respect to local image features such as texture, color, edges etc. Supervised segmentation methods typically extract specific regions from images through user interaction or from manually generated training images [1, 2, 3, 4, 5]. For medical images, obtaining ground-truth is a cumbersome process process, and is subject to intra and inter-operator variability. Several approaches have been proposed in literature for performing unsupervised image segmentation [6, 7, 8, 9]. Unsupervised segmentation is particularly useful for region-based image retrieval applications that derive "similar" regions/classes of unknown categories from a large database of images. Our objective for performing unsupervised segmentation is to automatically derive semantically relevant regions from medical images and create a visual ontology for the iMedline project at the National Library of Medicine (NLM) [10]. Since there can be multiple ways of partitioning an image, our goal is to sequentially generate hierarchical partitions using local image features. In this paper, unsupervised segmentation has been performed on low-resolution images to derive coarse segmentations from images. In future, we would like to incorporate automatic parameter selection based on top-down information to derive detailed regions from hierarchical image partitions.

Here, we explore the use of cellular automata (CA) for unsupervised image segmentation. This follows from evidence in literature where CA have been used to perform a wide variety of image processing tasks such as noise removal, edge detection, morphology detection, ridge-valley detection, convex hull detection etc. [11]. The proposed unsupervised grow-cut method is motivated by benefits offered by the interactive grow-cut algorithm [12] which offers reasonably robust segmentation but is dependent on user input and is also limited by the number of classes defined in the interaction. In the traditional method, a user can label some pixels in an image as "true" (+1) and some others as "false" (-1). Starting from the labeled pixels, the grow-cut algorithm propagates the labels using color features to derive an optimal cut (or segmentation) for a given image. However, the algorithm is dependent on the correctness of user-marked labels. To overcome this limitation and automate the process, we have developed the unsupervised grow-cut (UGC) algorithm, which can automatically identify the number of classes and the class boundaries based on local image features present in an image. Starting from random number of labels, the value for each cell (image pixel) is updated using the information of its current state and its neighbors. Cellular automata are a type of complex adaptive system in which, small locally operating units act in unison to produce a global emergent behavior. Cellular automata were introduced by von Neumann and Ulam and simulate the biological process of self-reproduction [13]. Other biologically-inspired computation systems such as Self-Organizing Feature Maps (SOFM) [14], Particle Swarm Optimization (PSO) [15], and Genetic Algorithms (GA) [16], etc. have been successfully used for medical image processing. Cellular-automata based interactive image segmentation was introduced fairly recently by Vezhnevets, and Konouchine [12] and used for interactive medical image segmentation by Kim et al. [17].

This paper is organized as follows. Section 2 describes the proposed algorithm in detail along with its relationship with other existing unsupervised segmentation algorithms: Mean Shift method and the Normalized Cut-based image segmentation method. The contribution of this research work is also presented in section 2. The description of datasets used and the results obtained are discussed in section 3. The future work and conclusions are described in section 4.

II. PROPOSED METHOD: UNSUPERVISED GROWCUT

Cellular automata are dynamic systems that evolve on a discrete grid of cells that interact locally to produce information at a global scale. In general, any discrete system undergoing deterministic local interactions can be modeled as a cellular automaton. Cellular automata computation is considered to be "brain-like" and modeled on computations performed by neurons in the human visual cortex. Typically, a CA consists of some pre-defined state-transition rules, which determine the value of each cell based on the value of its neighborhood cells. In case of image segmentation, the state transition rules are applied iteratively to update the value of each pixel in an image. A cellular automaton is defined using a triple (S, N, δ), where S is the state, N is the neighborhood and δ is the state transition function (rule). In the case of image segmentation, S is the class-label associated with a pixel. The neighborhood (N) of a pixel p is defined on $\mathbf{R}^2$ as

$$N = B(p;r) \bigcap \mathbf{Z}^2, \tag{1}$$

Where $B(p;r)$ is a ball (or circle) centered at pixel p with radius $r \geq 1$. Both the von Neumann (4-connected) or Moore (8-connected), neighborhoods may be defined for a two-dimensional cellular automaton. The state transition function defines the state of the central cell at time t+1 with respect to the values of cells in its neighborhood at time t.

The unsupervised grow-cut algorithm incorporates a CA-based framework using low-level image features such as color/gray-level pixel intensity values to derive state-transition rules. The initial labels and the number of seed points are derived randomly from the space of positive integer values. The state of each cell is given by a 3-tuple (l, θ, I), where l is the label, $\theta \in [0,1]$ is the strength of the cell and I is the pixel intensity. The strength of a cell is a function of image features and is used to define the state transition function for updating the labels of cells at each time step. Pixels with initial random labels are assigned the cell strength 1. The state transition rule between two pixels p and q is defined using a monotonically decreasing function g as

$$g\left(\left|I_p - I_q\right|\right) \cdot \theta_q > t \ ,$$

$$\text{where, } g(x) = 1 - \left(\frac{x}{\max|I|}\right). \tag{2}$$

Currently, the threshold parameter t, which determines the level of quantization of pixel intensity values, is assigned manually. Automatic parameter selection using top-down information is a subject of future work. In addition, equivalence classes are constructed corresponding to each label, which are updated when two labels merge. The algorithm is iterated until the local label update stops occurring. For a given image $\mathbf{X}$, the equivalence class for pixels p with labels l is defined as:

$$[l] = \{p \in \mathbf{X} \mid p \sim l\} \ . \tag{3}$$

The algorithm to update cell labels l and strengths θ is described below:

UGC update rule

```
// For each cell...
  for ∀p ∈ X
// Copy previous state
      l^{t+1} = l^t;
      θ^{t+1} = θ^t;
// neighbors try to attack current cell
      for ∀q ∈ N(p)
          if g(|I_p − I_q|)·θ_q > t AND l_p ≠ l_q
              l_p^{t+1} = l_q^t ;
              θ_p^{t+1} = g(|I_p − I_q|)·θ_q^t ;
              // create equivalence class
              if l_p^t ≠ 0
                  [l_p^{t+1}] = {p ∈ X | p ~ l_p^t}
              end if
          end if
      end for
  end for
```

A. Computational complexity

Since the UGC algorithm is dependent on state transitions occurring at local neighborhoods, the computational complexity is O(s*n), where s is the number of original seed points and n is the size of the local neighborhood. This is very advantageous because the computational complexity is independent of the image size or dimension of the feature space (color/texture etc.) that is used to determine the state transition rule.

B. Related Work

The UGC method is quite similar to the Mean Shift method of unsupervised image segmentation and the Normalized Cut-based image segmentation, which combine spatial information along with image features to perform segmentation. There are however some key differences between our approach and these two methods.

The mean shift algorithm works by averaging data points over a local neighborhood at successive iterations. It is a global optimization technique, in which data points are repeatedly shifted to the sample mean in a local neighborhood. In contrast, the data points remain unchanged in the case of the UGC algorithm. At each iteration, only the label matrix is updated using the cellular automata evolution rule. Instead of averaging over a window of points, the CA uses simple state transition rules to update the value of the central pixel in the local window. Since this state transition operation can be easily implemented using logic operations, implementing a cellular automata based segmentation on a N-dimensional image

becomes straightforward, which is not the case for mean shift clustering.

Normalized Cuts is a graph-based image partitioning algorithm, which uses a feature similarity measure and spatial proximity to derive optimal partitions of a weighted undirected graph. Unlike UGC, the number of desired partitions has to be specified by the user before performing segmentation.

C. Contributions

Incorporating cellular automata for unsupervised segmentation offers the following advantages:

1. The number of classes need not be specified before segmentation is performed.
2. Since cellular automata use simple state transition rules, it is extensible to other features. Currently pixel intensity values have been used as state transition rules, but other image features such as texture, edges, probability maps depicting relative positions etc. could be easily incorporated into this framework.
3. The computational complexity of the segmentation algorithm is independent of the image size or the number of image features.

III. Results

The algorithm was tested on a synthetic image, 300 natural images from the Berkeley dataset and 30 images from the ImageCLEFmed database. The results obtained from each of the datasets is described below.

A. Synthetic image segmentation

At first, the unsupervised grow-cut algorithm was tested on a synthetic image with four objects (Fig. 1(a)). Fig. 1(b) shows the initial seed points (labels) that were randomly assigned to the image. Panels (c) and (d) in Fig. 1 show regions found during the process of growing and merging of the labels using an equivalence relation to derive final number of classes and class boundaries. Fig. 2(a) shows the number of classes derived by UGC as a function of the number of initial seed points. The figure suggests that when a reasonably good size of initial seed points is chosen, the algorithm consistently returns the same number of classes. Fig 2(b) depicts that the execution time of the algorithm is a function of the number of original seed points. This is expected because the algorithm is $O(s*n)$ as discussed previously. For this experiment, a fixed-size local neighborhood of 3x3 was chosen.

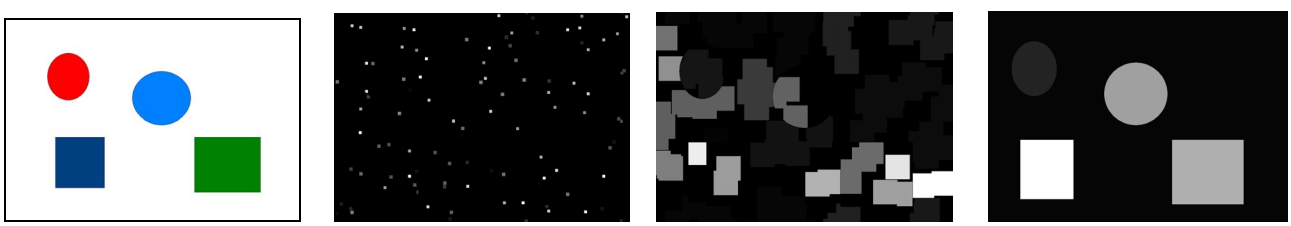

Figure 1. (a): Original image; (b): Initial random seed points with labels; (c) – (d): UGC grows and merges the random labels until it finds the desired classes based on color feature; (d) Segmented image with 5 labels. One for each object, and the background.

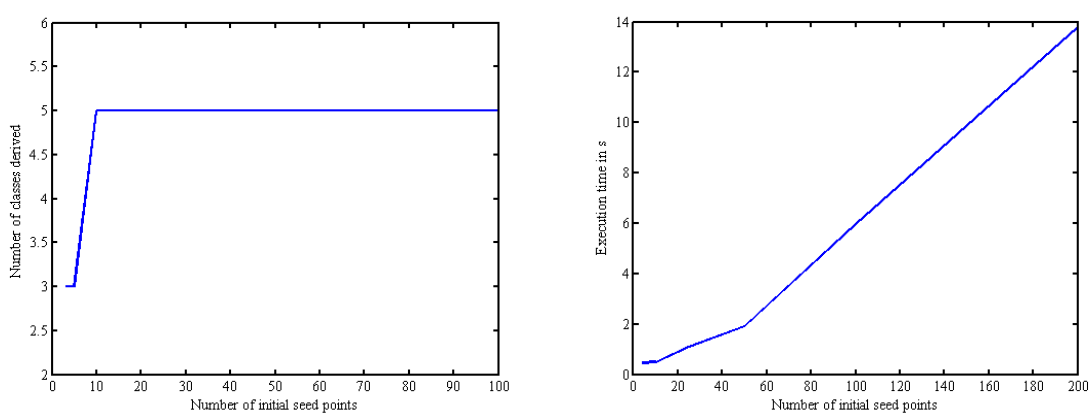

Figure 2. (a):The number of classes derived by UGC as a function of the number of initial seed points.(b): The execution time of UGC as a function of the number of initial seed points

B. *Natural image segmentaiton*

Figure 3. Segmentation outcome from UGC on four natural images from the Berkeley database.

Unsupervised segmentation using UGC was then performed on the Berkeley dataset consisting of 300 natural images. This dataset was chosen, because it consists of ground truth segmentation and could be used to compare the segmentation outcome of UGC against the Normalized Cuts method [1] and the Mean Shift algorithm [2] on four evaluation measures:

1. The Probabilistic Rand Index (PRI), which compares the segmentation outcome and the ground truth and counts the fraction of pairs of pixels that are correctly labeled.

2. The Variation of Information (VoI), which is defined as the average conditional entropy of one segmentation result given the other.

3. The Global Consistency Error (GCE), which measures the extent to which a segmentation result can be viewed as a refinement of the other at different scales.

4. The Boundary Displacement Error (BDE), which measures the average displacement error of boundary pixels between two segmented images. It computes the distance between a pixel in one boundary image and the closest pixel in the other boundary image.

Figures 3 shows segmentation outcome from UGC on four natural images from the Berkeley database. Table 1 shows the average evaluation measures derived from the segmentation outcomes from the 300 images. The table shows that the segmentation results from UGC are comparable with the Mean Shift and the Normalized cuts methods of image segmentation.

TABLE I. SEGMENTATION OUTCOME OF THE UGC COMPARED WITH THE NCUTS AND MS APPROACH.

	PRI	**VoI**	**GCE**	**BDE**
Human	0.8574	1.1	0.0797	4.99
Mean Shift	0.76	2.4	0.23	9.7
NCuts	0.72	2.9	0.22	9.6
UGC	0.68	2.3	0.27	9.9

C. *Medical image segmentation*

The UGC algorithm was then compared with the mean shift algorithm (MS) and the normalized-cut method (NCut) of unsupervised segmentation on a subset of 30 MRI images of the lung and brain derived from the ImageCLEFmed database [18] from Oregon Health & Science University (OHSU). The parameters used for each algorithm were: UGC (threshold=0.95, initial labels=100), MS (spatial bandwidth=8, range bandwidth=4, minimum region area=50), NCut (number of final labels=10). Figures 4 and 5 show segmentation outcomes on 10 lung and brain images from the ImageCLEFmed database. The leftmost panel in these figures shows the original image resized to a lower resolution of 64x64 pixels. The next panel shows segmentation outcome using UGC. The other two panels show segmentation outcomes using the MS and the NCut methods respectively. Since the imageCLEFmed database does not have ground truth segmentation, quantitative evaluation of segmentation results could not be performed for this dataset. A visual inspection of the segmentation outcomes from all the images shows that the results of UGC are comparable with the other methods. Figure 6 depicts the stopping criteria of the UGC by plotting the number of pixels updated in the label matrix as a function of the number of iterations for the top two images of Figure 4.

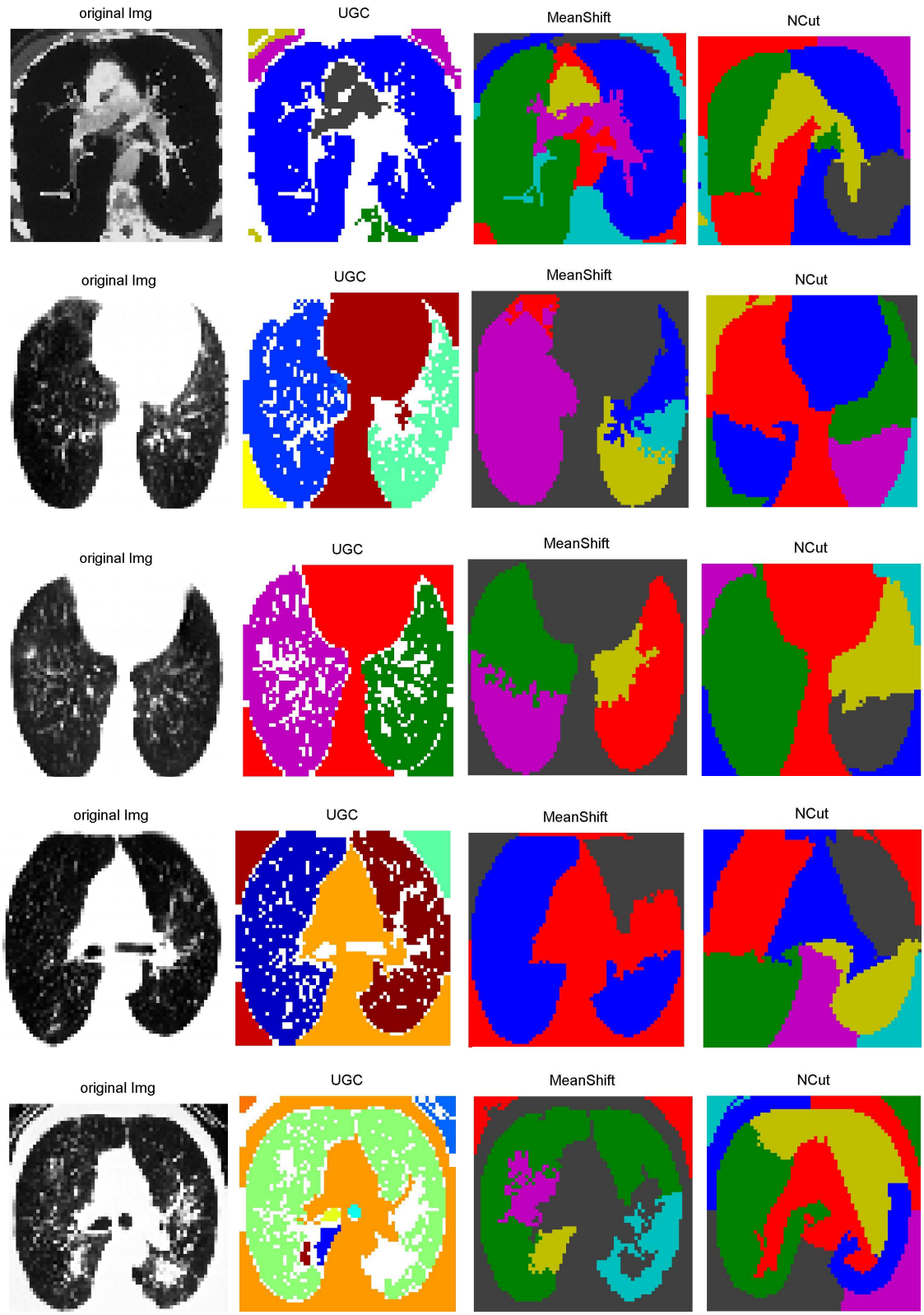

Figure 4. (a):Original image 64x64. (b): Segmentation outcome using UGC. (c): Segmentation outcome using MS. (d): Segmentation outcome using NCut. The pseudo color images were generated using *label2rgb* command in Matlab®.

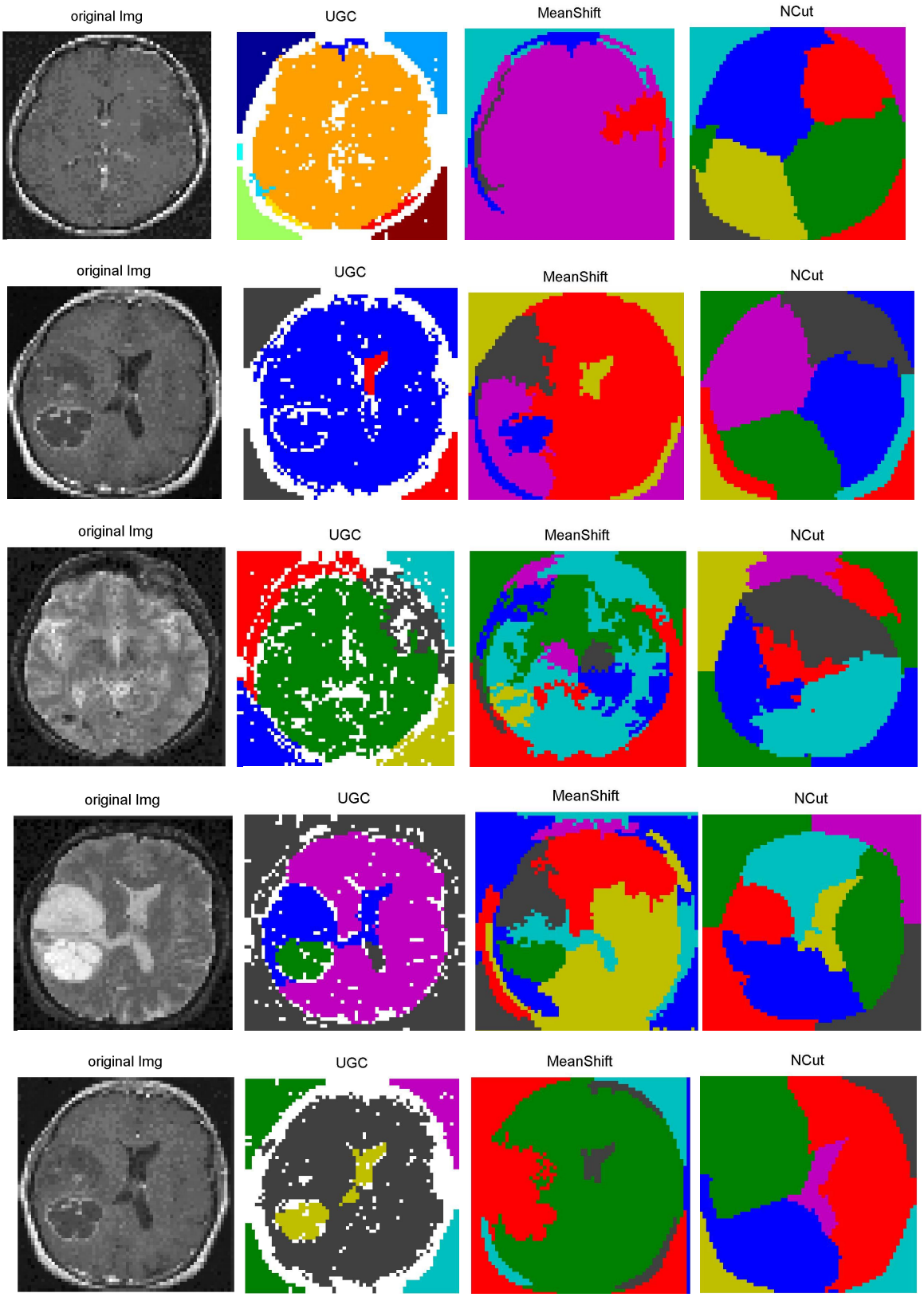

Figure 5. (a):Original image 64x64. (b): Segmentation outcome using UGC. (c): Segmentation outcome using MS. (d): Segmentation outcome using NCut. The pseudo color images were generated using *imagesc* command in Matlab.

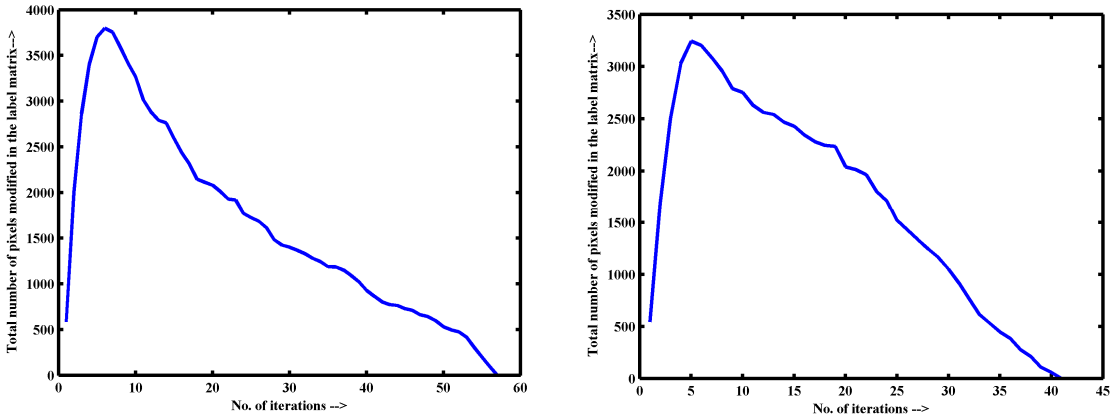

Figure 6. The stopping criteria of the UGC depicted by plotting the number of pixels updated in the label matrix as a function of the number of iterations for top two images of Figure 4.

IV. Conclusion and Future Work

We present an unsupervised image segmentation technique using cellular automata where the number of classes and class labels are automatically determined by the algorithm at run time. The method has been compared with the mean shift method and the normalized cut method for unsupervised image segmentation. The segmentation outcomes of our CA-based method are comparable to the results of existing unsupervised segmentation methods. Segmentation using cellular automata is advantageous for the following reasons: efficient parallel implementation possible; computational complexity is independent of dimension of the image; extensibility to other features.

As mentioned previously, the algorithm parameter is currently set manually by the user. Since this algorithm would be used for image retrieval application, automatic selection of parameters using text-based top down information would be performed in future to derive relevant details from images.

Since a unique set of deterministic state transition rules is used to segment each image, segmentation and region characterization may be performed in a single step using this method. Validation of state transition rules as a means to quantitatively characterize anatomical regions present in medical images is another subject of future work for the iMedline medical image retrieval project at the National Library of Medicine (NLM).

Acknowledgment

This research was supported by the Intramural Research Program of the National Institutes of Health (NIH), National Library of Medicine (NLM), and Lister Hill National Center for Biomedical Communications (LHNCBC). We also thank the CLEF [18] organizers and the Berkeley computer vision group [19] for making the database available for experiments.

References

[1] C. Rother, V. Kolmogorov, A. Blake, “GrabCut: interactive foreground extraction using iterated graph cuts,’ SIGGRAPH , vol. 23, pp. 309-314, 2004.

[2] Y. Boykov, M. P. Jolly, “Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images,” Proc. IEEE Intl. Conf. on Comp. Vis., 2001.

[3] V. Lempitsky, P. Kohli, C. Rother, T. Sharp, “Image segmentation with a bound-ing box prior,” Proc. IEEE Intl. Conf. on Computer Vision, 2009.

[4] F. Schro, A. Criminisi, A. Zisserman, “Object class segmentation using random forests,” Proceedings of the British Machine Vision Conference, 2008.

[5] M.P. Kumar, P. H. S. Torr, A. Zisserman, “OBJ CUT,” Proceedings IEEE Conference on Comp. Vis. Patt. Recognition, pp. 18-25, 2005.

[6] L. Cao, F. F. Li, “Spatially coherent latent topic model for concurrent segmentation and classication of objects and scene,” Proc. IEEE Intl. Conf. on Comp. Vis., 2007.

[7] S. Jianbo, and J. Malik, “Normalized cuts for image segmentation,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22(8), pp. 888-905, August 2000.

[8] C. Yizong, "Mean shift, mode seeking, and clustering," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.17, no.8, pp. 790-799, Aug 1995.

[9] J. Winn and N. Jojic, LOCUS: Learning Object Classes with Unsupervised Segmentation, Proc. IEEE Intl. Conf. on Computer Vision (ICCV), Beijing 2005.

[10] D. Demner-Fushman, S. K. Antani, M. Simpson, Md. M. Rahman, “Combining Text and Visual Features For Biomedical Information Retrieval,” Internal Technical Report, NIH. (http://archive.nlm.nih.gov/pubs/reports/techreport.php)K. Jarrah, M. Kyan, S. Krishnan, and L. Guan, “Computational intelligence techniques and their applications in content-based image retrieval,” Proceedings of IEEE International Conference on Multimedia and Expo., pp. 33-36, 2006.

[11] P. L. Rosin, “Image processing using 3-state cellular automata,” Comput. Vis. Image Underst., vol. 114 (7), pp. 790-802, 2010.

[12] V. Vezhnevets, and V. Konouchine, “GrowCut-interactive multi-label ND image segmentation by cellular automata,” Proceedings of Graphicon., pp. 231–234, 2006.

[13] Von Neumann, J. 1966. Theory of Self-Reproducing Automata. University of Illinois Press. Ed. and Completed by A. Burks.

[14] K. Jarrah, M. Kyan, S. Krishnan, and L. Guan, “Computational intelligence techniques and their applications in content-based image

retrieval," Proceedings of IEEE International Conference on Multimedia and Expo., pp. 33-36, 2006.

[15] N. Oka, and K. Kameyama, "Relevance tuning in content-based retrieval of structurally-modeled images using Particle Swarm Optimization," Proceedings of IEEE Symposium on Computational Intelligence for Multimedia Signal and Vision Processing, pp. 75-82, 2009.

[16] T. Zhao, J. Lu, Y. Zhang, Q. Xiao, "Feature Selection Based on Genetic Algorithm for CBIR," Congress on Image and Signal Processing, vol. 2, pp. 495-499, 2008.

[17] E. Kim, T. Shen, X. Huang, "A Parallel Cellular Automata with Label Priors for Interactive Brain Tumor Segmentation", Proceedings of the 23rd IEEE Intl. Symp. on Computer-Based Medical Systems, CBMS 2010.

[18] H. Műller, J. Kalpathy-Cramer, C. E. Kahn Jr., W. Hatt, S. Bedrick and W. Hersh, "Overview of the ImageCLEFmed 2008 Medical Retrieval Task," 8th Workshop of the Cross-Language Evaluation Forum (CLEF 2007), Proceedings of LNCS, 5152, 2008.

[19] 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", Proc. 8th Int'l Conf. Computer Vision, vol. 2, pp. 416-423, July, 2001.