

Computationally Efficient Segmentation Model for Collection of Images

G.SARANYA

Student of Master of Electronics and Communication Engineering,
Sri Manakula Vinayagar Engineering College, Puducherry.
saranya.sekarifet@gmail.com

L.M.VARALAKSHMI

Associate Professor, Department of Electronics and Communication Engineering,
Sri Manakula Vinayagar Engineering College, Puducherry.
varalakshmi_1@yahoo.co.in

R.DEEPA

Student of Master of Electronics and Communication Engineering,
Sri Manakula Vinayagar Engineering College, Puducherry.
deepa6121989@gmail.com

Abstract— A semisupervised optimization model for determining an efficient segmentation of many input images is proposed in this paper. The advantage of this model is twofold. Firstly, the segmentation is highly controllable as the portion chosen for segmentation can be specified by providing the labeled pixels in images for the model either offline or interactively. Secondly, the model requires only minimal tuning of model parameters during the initial stage. Once initial tuning is done, it can be used to automatically segment a large collection of images that are distinct but share similar features. It is proposed to conduct extensive experiments on various collections of biological images, it will be established that the model proposed is quite computationally efficient and effective for segmentation.

Keyword- Biological image segmentation, Image segmentation, Interactive, Microscopic images, Multiple images.

1. INTRODUCTION

Image Segmentation is the process of partitioning a digital image into multiple segments and used to locate objects and boundaries (lines, curves) in images. It is used in many areas, including computer vision, computer graphics, and medical imaging etc. Types of image segmentation are Fully automatic image segmentation and Semi-automatic image segmentation. Fully automatic image segmentation has many intrinsic difficulties and is still a very hard problem. For example, it is very often that an image can have much segmentation that is meaningful. Fields like medical or biomedical imaging, objects of interest (OOIs) are often badly defined and even sophisticated automatic segmentation algorithms often fail. Moreover, in cell segmentation in microscopy images and organ segmentation in medical images, the kind of objects and segmentation of interest are known in advance. It is therefore tempting to design segmentation methods that allow the user to specify what the user wants.

For these situations, the only possibility until recently was to replace automatic methods by interactive (or manual) ones, where a lot of interaction between the user and the image is necessary, either to draw the contours of OOIs .It is always very tedious. So, it is replaced by semi-automatic image segmentation, with a very limited amount of user interaction. Several types of semi-automatic methods have been suggested: intelligent scissors, methods based on user steered image segmentation paradigms and methods based on the concept of fuzzy connectedness.

The segmented objects are clustered to retrieve the original image. In clustering there are three type of clustering ie., supervised, unsupervised and semi-supervised. Semi-supervised clustering is introduced to cover some drawbacks of clustering (unsupervised learning) and classification (supervised learning), such as production of non acceptable clusters or sometimes finding multiple grouping of data in the clustering process. In this situation semi-supervised clustering could be a good choice. Semi-supervised clustering uses some side-information to cover the categorization goal. This side-information could be the similar pairs from input data or information that indicates membership of the data items to specific clusters. This side-information usually has the pair-wise (must-link and cannot-link constraints) form in most studies. Must-link constraints impose data on the same cluster but cannot-link constraints impose them on different clusters.

In semiautomatic segmentation, the user marks some sample pixels from each class of objects. The computational algorithm then computes a classification of other pixels. This way, the resulting segmentation is highly controllable by the user and thereby eliminates much ambiguity in defining a partition. Because of this property it is used in medical field [2] & [3].

Initially optimization model is available for single class only. Later an optimization-based two-class segmentation model [4] is developed, in which an optimal class membership function is computed through the minimization of a quadratic cost function with user-supplied samples as linear constraints. The basic idea is that two pixels should have similar membership if they are either geometrically similar or photometrically similar or both. The results are quite impressive. The model was later extended [5] in to handle the multiple-class problem. Some effective numerical optimization methods and fundamental theoretical properties of the model were studied [6], [7] & [8].

Single-image optimization models were extended to the multiple-image for image retrieval. The various clustering techniques are *k-means* and *support vector Machine*. K means is an unsupervised method used to group the objects based on attributes/features into K number of group. The grouping is done by minimizing the sum of squares of distances between data and the corresponding cluster centroid. The drawback is user has to specify the number of clusters in advance, unable to handle noisy data and it is not suitable to discover clusters with non-convex shapes. Support Vector Machines is a supervised method, which is well-suited for aspect-based recognition and color-based classification. SVM is widely used in object detection & recognition, Text recognition, etc the drawback is it's sensitive to noise, it considers only two classes and image classification problem exist.

The outline of this paper is as follows. In Section 2, the proposed model and the properties are discussed. In Section 3, the experimental result of the proposed model is shown. In Section 4, some concluding remarks are given.

2. PROPOSED SCHEME

In this section, the formulation of the proposed model is stated. The two-image multiple-class case is illustrated in this section. This two-image model can be used to segment a collection of images one at a time. The generalization to multiple-image multiple-class case is clear.

2.1 Optimization Model

Let u^s for $s = 1, 2$ be two given multichannel images. Their sizes are not necessarily the same. Let Y^s be the set of all pixels in image u^s . Let Ω^s be the set of all unlabeled pixels in image. Let Γ^s be the set of pixels in image u^s labeled to one of the M classes by the user. Thus $Y^s = \Omega^s \cup \Gamma^s$ which allowing both labeled and unlabeled pixels contained in the image. The set of labeled pixels Γ^s is divided into $\Gamma^{s/1}, \dots, \Gamma^{s/M}$, where $\Gamma^{s/m}$ is the set of pixels that are labeled with class m , for $m=1, \dots, M$. s' is an index referring to an image different from the image indexed by s . For each pixel $i \in Y^s$ and each pixel $j \in Y^{s'}$, let $w_{i,j}^{s,t} \geq 0$ be a similarity between the pair of pixels, for $s, t=1, 2$. When $t=s$, the similarity $w_{i,j}^{s,t}$ is computed within image u^s ; when $t=s'$, the similarity is computed across two images. For each $i \in Y^s$, the similarity scores are normalized as shown in (1)

$$\sum_{j \in Y^s} w_{i,j}^{s,s} + \sum_{j \in Y^{s'}} w_{i,j}^{s,s'} \equiv 1 \quad (1)$$

For each pixel $i \in Y^s$, let $N_i^{s,t} \subset Y^t$ be a set of pixels in image, which is called the *neighbor* of i in u^t . For each $i \in Y^s$, let $\alpha_i^{s/m} \in [0, 1]$ be the degree of membership of pixel $i \in Y^s$ to class m . It is required that $\sum_{m=1}^M \alpha_i^{s/m} = 1$. It is also denote by the vector $(\alpha_i^{s/m})_{i \in Y^s}$. The basic idea is that the memberships of similar

pixels should be similar. For each unlabeled pixel $i \in \Omega^s$, the membership to class m inferred from its neighbors is the weighted average as shown in (2)

$$\sum_{j \in N_i^{s,s}} w_{i,j}^{s,s} \alpha_j^{s/m} + \sum_{j \in N_i^{s,s'}} w_{i,j}^{s,s'} \alpha_j^{s'/m} \quad (2)$$

$$\min_{\{\alpha^{s/m}\}_{s,m}} \sum_{m=1}^M \sum_{s=1}^2 \sum_{i \in \Omega^s} \left[\alpha_i^{s/m} - \sum_{j \in N_i^{s,s}} w_{i,j}^{s,s} \alpha_j^{s/m} - \sum_{j \in N_i^{s,s'}} w_{i,j}^{s,s'} \alpha_j^{s/m} \right]^2 \tag{3}$$

Subject to the constraints,

$$0 \leq \alpha^{s/m} \leq 1 \quad \text{and} \quad \sum_{m=1}^M \alpha^{s/m} = 1 \tag{4}$$

For s= 1, 2 and m=1,...,M and the boundary condition $\alpha_i^{s/m} = 1$, for $i \in \Gamma^{s/m}$ and $\alpha_i^{s/m} = 0$, for $i \in \Gamma^s \setminus \Gamma^{s/m}$.

The objective function in (2) can be compactly written in matrix form as shown in (5)

$$J(\alpha^1, \dots, \alpha^M) = \sum_{m=1}^M \|DA\alpha^m\|_2^2 \tag{5}$$

2.2 Similarity Measures

Two kinds of similarity measures are, geometric and photometric, are considered. The former is based on pixel locations, whereas the latter is based on color features.

For each pixel $i \in Y^s$, its *geometric neighbor* $G_i^{s,s} \subset Y^s$ is defined in (6)

$$G_i^{s,s} := \{j \in Y^s : 0 < \|i - j\|_\infty \leq r_g\} \tag{6}$$

where $r_g > 0$ is a constant controlling the size of the window, and $\|\cdot\|_\infty$ is the vector maximum norm. We often set $r_g = 1$ so that a 3*3 window around pixel i is used.

The *geometric similarity* $g_{i,j}^{s,s}$ is defined in (7)

$$g_{i,j}^{s,s} := \begin{cases} c e^{-\frac{\|i-j\|_2^2}{\sigma_i^2}}, & \text{if } j \in G_i^{s,s} \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

where c is a normalization constant such that, $\sum_{j \in Y^s} g_{i,j}^{s,s} = 1$, and σ_i^2 is computed as the sample variance of

the geometric locations within $G_i^{s,s}$. For each pixel $i \in Y^s$, let F_i be its feature vector.

The *within-image photometric neighbor* $P_i^{s,s} \subset Y^s$ is defined to be the top 4 pixels within the 17*17 window around pixel whose feature vectors are nearest to F_i . Using a larger window size allows us to reduce error. The *within-image photometric similarity* is defined in (8)

$$p_{i,j}^{s,s} := \begin{cases} c e^{-\frac{\|F_i - F_j\|_2^2}{\rho_i^2}}, & \text{if } j \in P_i^{s,s} \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

where, ρ_i^2 is computed as the sample variance of the photometric features within $P_i^{s,s}$.

For efficient computation of the *across-image photometric neighbor* $P_i^{s,s'} \subset Y^{s'}$ the top 4 labeled pixels in $S \subset \Gamma^{s'}$ is considered, whose feature vectors are nearest to F_i . Here, S is a random sample of $\Gamma^{s'}$ such that it contains an equal number of pixels from $\Gamma^{s'/1}$ and $\Gamma^{s'/2}$. The *across-image photometric similarity* is defined in (9)

$$p_{i,j}^{s,s'} := c e^{-\frac{\|F_i - F_j\|_2^2}{\theta_i^2}}, \text{ if } j \in P_i^{s,s'} \tag{9}$$

Where c is a normalization constant such that $\sum_{j \in \gamma^{s,s'}} p_{i,j}^{s,s'} \equiv 1$, and θ_i^2 is computed as the sample variance of the photometric features within $P_i^{s,s'}$.

The *within-image neighbor* and the *across-image neighbor* are defined in (10) and (11) as

$$N_i^{s,s} := G_i^{s,s} \cup P_i^{s,s} \tag{10}$$

$$N_i^{s,s'} := P_i^{s,s'} \tag{11}$$

The *combined within-image similarity* and *combined across-image similarity* is defined in (12)&(13)

$$w_{i,j}^{s,s} := \frac{1}{1+\mu} \left(\frac{1}{1+\lambda} g_{i,j}^{s,s} + \frac{\lambda}{1+\lambda} p_{i,j}^{s,s} \right) \tag{12}$$

$$w_{i,j}^{s,s'} = \frac{\mu}{1+\mu} p_{i,j}^{s,s'} \tag{13}$$

Where, $\lambda > 0$ is a tuning parameter controlling the weight between geometric and photometric similarities, and $\mu > 0$ is a tuning parameter controlling the weight between within- and across-image similarities.

Three methods for computing across-image similarity are

- i) All pixel: All the pixels in the image is taken. (labeled and unlabeled pixels)
- ii) Random pixel: A set of randomly selected pixels in u^1 where, u^1 - labeled pixels in the image
- iii) SIFT Keypoints: Scale-invariant feature transform (SIFT) is an algorithm in computer vision to detect and describe local features in images. Applications include object recognition, image stitching, video tracking, and individual identification of wildlife and match moving.

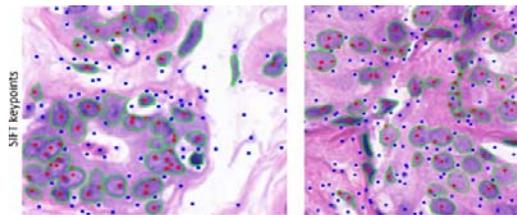


Figure 1. Scale Invariant Feature Transform keypoints detection

In Fig. 1 the red and blue dots are SIFT keypoints served as the photometric neighbour set for the nucleus part and the background part.

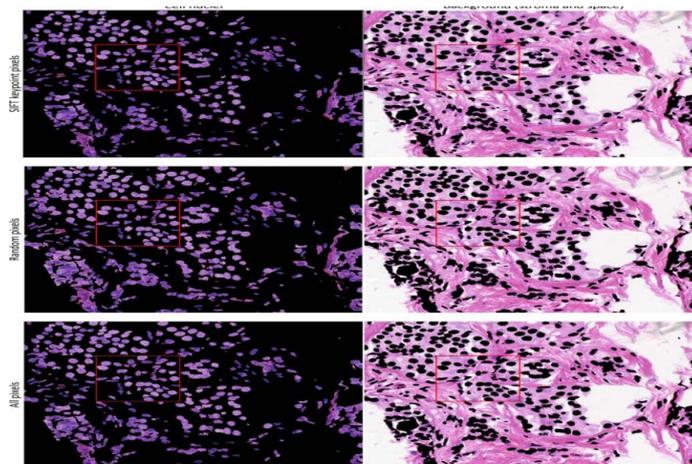


Figure 2. Comparison of all the three methods

To find the performance between all the three methods, the All pixels method is the baseline. The percentage symmetric difference between random pixels and SIFT keypoints method have less than 3% for both the images as shown in Fig. 2. Random pixels and SIFT keypoints method is 3-5 times faster than All labeled pixels. The running time is almost same when the same numbers of pixels are selected in both random pixels and SIFT keypoints. By trying various percentage of selected points from 0.2% to 100%, the percentage

symmetric differences decreases from 3% to 0%.So the SIFT keypoints method is used in Multiple-image model.

2.3. Optimality Conditions

The objective function in (5) is differentiated and Lagrange multipliers for the constraints is introduced in (4), to shown that the optimality conditions are given in the linear systems as shown in (14),

$$\begin{aligned} \tilde{A}\alpha^m &= b^m \quad \text{for } m=1,\dots,M \\ \tilde{A} &= I - D^T DW \\ &= \begin{pmatrix} I - D_{\Omega^1}^T D_{\Omega^1} W^{1,1} & -D_{\Omega^1}^T D_{\Omega^1} W^{1,2} \\ -D_{\Omega^2}^T D_{\Omega^2} W^{2,1} & I - D_{\Omega^2}^T D_{\Omega^2} W^{2,2} \end{pmatrix} \\ b^m &= \begin{bmatrix} b^{i/m} \\ b^{2/m} \end{bmatrix} \end{aligned} \tag{14}$$

Each unlabeled pixel is connected to a labeled pixel through a sequence of directed edges, each of which connects a pixel to one of its neighbors in the same image or a different image. It shows that the solution is nonsingular and unique. If the matrix size is small, then linear systems can be efficiently solved by Gaussian elimination. However, if the image size is larger, preconditioned iterative methods [5] are used.

2.4. Application to a Collection of Images

Suppose u^1 contains some manually labeled pixels while other images are unlabeled. To segment a collection of images, we apply the multiple-image model to u^1 and one other image (called u^2) at a time. That implies $\Gamma^1 \neq 0$ and $\Gamma^2 = 0$

A simple way to apply the model is to let $P_i^{1,2} = 0$ for all $i \in Y^1$, so that $W^{1,2} = 0$, and let $w^{1,1} = \frac{1}{1+\lambda} G^{1,1} + \frac{\lambda}{1+\lambda} P^{1,1}$. In this case, the matrix

is a block upper triangular as shown in (15)

$$\tilde{A} = \begin{pmatrix} I - D_{\Omega^1}^T D_{\Omega^1} W^{1,1} & 0 \\ -D_{\Omega^2}^T D_{\Omega^2} W^{2,1} & I - D_{\Omega^2}^T D_{\Omega^2} W^{2,2} \end{pmatrix} \tag{15}$$

The parameters λ and μ are manually tuned to solve $\alpha^{1/m}$ and the first unlabeled image. Then, These values are used to segment all other images. Thus the tuning is quite easy for the collection that is used.

If there $K > 1$ in images that contain labeled pixels, then we can simply apply the model for $k+1$ images to segment one unlabeled image at a time. The model proposed is easy to be extended. For the single-image case, it show that α^m satisfies the strong maximum principle, which guarantees the strict inequalities $0 < \alpha^m < 1$ and the uniqueness of α^m [5]. For the multiple-image, if $W^{1,2} = 0$, then the weak maximum principle, which implies $0 \leq \alpha^m \leq 1$ only. However, the more important uniqueness of α^m still holds.

2.5. Computational Complexity

In multiple-image model, the computational costs is discussed in the following steps

Step 1) Compute $P^{1,1}$ (independent λ of and μ).

Step 2) Compute $W^{1,1}$ (dependent on λ) and solve the linear system $[I - D_{\Omega^1}^T D_{\Omega^1} W^{1,1}] \alpha^{1/m} = b^{1/m}$ for $m=1,2,\dots,M-1$.

Step 3) Compute $P^{2,2}$ and $P^{2,1}$ (independent of λ and μ).

Step 4) Compute $W^{2,1}$ and $W^{2,2}$ (dependent on λ and μ) and solve the linear system

$$[I - D_{\Omega^2}^T D_{\Omega^2} W^{2,2}] \alpha^{2/m} = D_{\Omega^2}^T D_{\Omega^2} W^{2,1} \alpha^{1/m} \quad \text{for } m=1,\dots,M-1.$$

During the initial tuning stage the parameters are tuned based on the labeled image and the first unlabeled image, steps 1 and 3 are need to be done only once, whereas steps 2 and 4 have to be repeated. In this experiments, steps 1 and 3 are often more time consuming than steps 2 and 4. If u^1 is fully labeled, then $\{\alpha^{1/1}, \dots, \alpha^{1/M}\}$ are known, and steps 1 and 2 can be skipped. Starting from the second unlabeled image, only steps 3 and 4 are performed, and no further tuning of parameters is done.

3. SIMULATION RESULTS AND PERFORMANCE ANALYSIS

The 5 retinal testing images are taken to segment one at a time using the proposed multiple-image model, with the image in Fig. 4 as a fully labeled sample. The input image is converted into grayscale image for easy analysis of labeled and unlabeled pixels. For single object in an image the threshold value is determined and then, labeled and unlabeled pixels are marked. First let's consider the single object retinal images as shown in Fig. 3.



Figure 3. Original retinal image

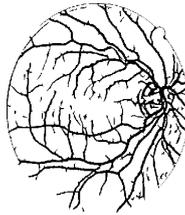


Figure 4. Labeled image

The similarity measures are calculated by comparing Fig. 3 with Fig. 4 and the required portion is extracted for the first image as shown in Fig. 5

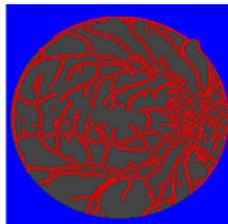


Figure 5. Segmentation of retinal image obtained by the Multiple-image model

The output of first image obtained by the multiple-image model is applied as a reference parameter for segmenting the remaining images automatically. The output obtained for the remaining images as shown in Fig. 6

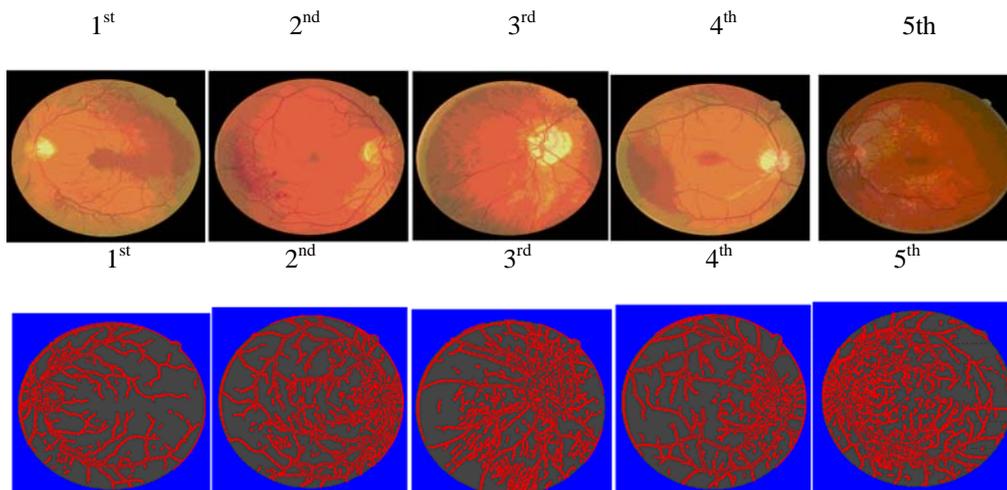


Figure 6. Segmentation of five of the testing retinal images obtained by the Multiple-image model

The Breast cancer cell image as shown in Fig. 7 are taken to segment one at a time using Multiple-image with the image in Fig. 8 as reference. Here multiple objects are there in an image so they have to be

segmented separately, then the required object has to be specified and then the clustering are done using the proposed multiple-image model.

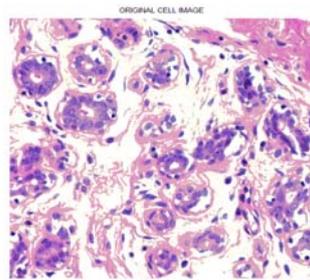


Figure 7. Original cell image

In this image there are 4 objects as shown Fig. 8. So, it is segmented and all the objects are separated and the required object ie., nuclei is selected as shown in Fig. 9.

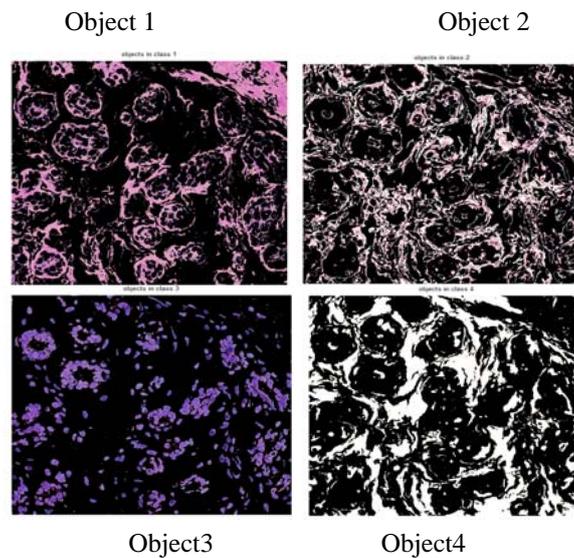


Figure 8. Multiple object in the image

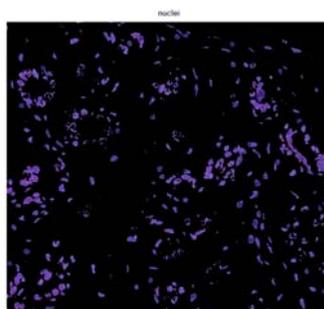


Figure 9. Required object in cell image

The similarity measures are calculated by comparing Fig. 7 with Fig. 9 and the required portion is extracted for the first image as shown in Fig. 10.

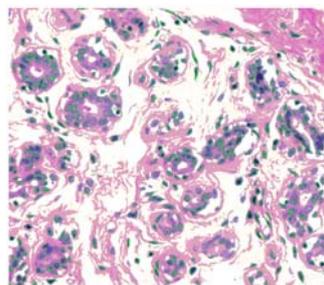


Figure 10. Cell image obtained by the Multiple-image model

The output of first image obtained by the multiple-image model is applied as a reference parameter for segmenting the remaining images automatically. The output obtained for the remaining images as shown in Fig. 11.

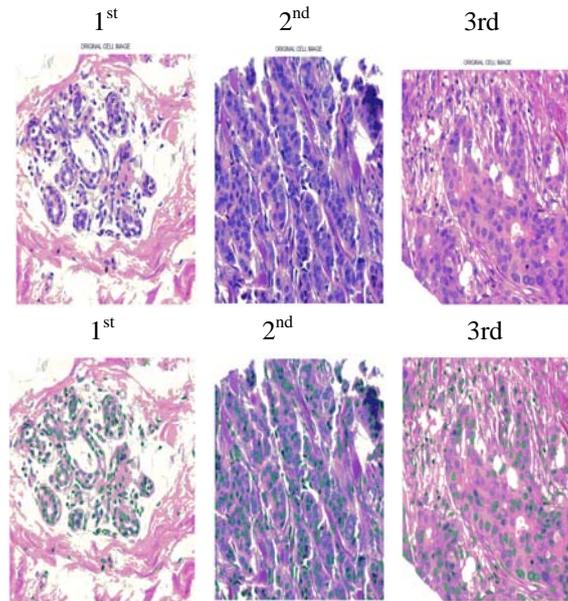


Figure 11. Segmentation of three of the testing cell images obtained by the Multiple-image model

Performance Analysis

The performance analysis for Multiple Image clustering technique is computed. It is found that the Multiple-image model shows better accuracy and F-measures when compared to other technique as shown in Fig. 12 and Fig. 13.

i) The Accuracy is calculated as

$$\text{Accuracy} = \frac{\text{The No. of correctly classified pixels}}{\text{Total No. of pixels}}$$

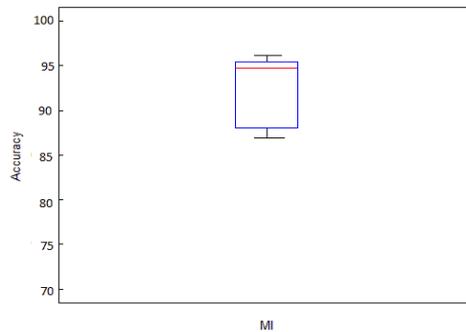


Figure 12. Accuracy

The accuracy for multiple image technique varies from 85% to 95%. While for other techniques (iterative and automatic clustering) the accuracy is less than 85 %.

ii) F-measure is calculated as

$$F = \frac{2 \text{ precision} * \text{recall}}{\text{precision} + \text{recall}}$$

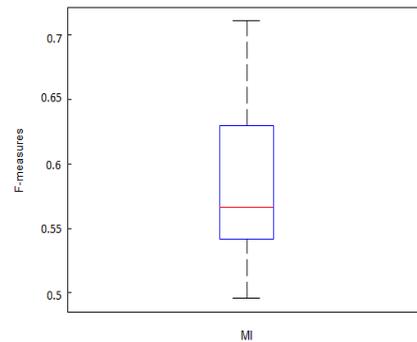


Figure 13. F-measures

The F-measures for multiple image technique varies from 0.54 to 0.63. While for other techniques (iterative and automatic clustering) the accuracy is less than 0.6.

4. CONCLUSION

A Semiautomatic optimization model for segmentation of multiple images is developed. The model has a quadratic objective function and linear constraints. Due to the discrete maximum/ minimum principles, the optimality conditions simply boil down to solve the linear systems. In our applications, the two parameters can be easily tuned. Once initial tuning is done, the setup can be used to segment all other images within the collection automatically. The quality of the results is also high. However, it relies on the logical assumption that the different classes can be separated in the feature space and that the user-supplied samples can represent each class well.

REFERENCES

- [1] Michael K. Ng, and Andy M. Yip, "A Semisupervised Segmentation Model for Collections of Images", *IEEE transactions on Image processing*, vol.21,no.6, June 2012.
- [2] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1124–1137, Sep. 2004.
- [3] A. Protiere and G. Sapiro, "Interactive image segmentation via adaptive weighted distances," *IEEE Trans. Image Process.*, vol. 16, no. 4, pp. 1046–1057, Apr. 2007.
- [4] J. Guan and G. Qiu, "Interactive image segmentation using optimization with statistical priors," in *Proc. ECCV, Graz, Austria, 2006*.
- [5] M. K. Ng, G. Qiu, and A. M. Yip, "Numerical methods for interactive multiple-class image segmentation problems," *Int. J. Imag. Syst. Technol.*, vol. 20, no. 3, pp. 191–201, Sep. 2010.
- [6] F. Wang, C. Zhang, H. Shen, and J. Wang, "Semi-supervised classification using linear neighborhood propagation," in *Proc. CVPR, 2006*, pp. 160–167.
- [7] O. Delalleau, Y. Bengio, and N. Le Roux, "Efficient non-parametric function induction in semi-supervised learning," in *Proc. 10th Int. Workshop AI Stat.*, 2005, pp. 96–103.
- [8] J. Guan and G. Qiu, "Modeling user feedback using a hierarchical graphical model for interactive image retrieval," in *Advances in Multimedia Information Processing*, ser. Lecture Notes in Computer Science, H.H.-S. Ip, H. Leung, O. C. Au, M.-T. Sun, W.-Y. Ma, and S.-M. Hu, Eds. Berlin, Germany: Springer-Verlag, 2007, pp. 18–29.