Segmentation of Touching Cell Nuclei Using a Two-Stage Graph Cut Model

Ondřej Daněk[1], Pavel Matula[1], Carlos Ortiz-de-Solórzano[2], Arrate Muñoz-Barrutia[2], Martin Maška[1], and Michal Kozubek[1]

[1] Centre for Biomedical Image Analysis, Faculty of Informatics
Masaryk University, Brno, Czech Republic
xdanek2@fi.muni.cz
[2] Center for Applied Medical Research (CIMA)
University of Navarra, Pamplona, Spain

Abstract. Methods based on combinatorial graph cut algorithms received a lot of attention in the recent years for their robustness as well as reasonable computational demands. These methods are built upon an underlying Maximum a Posteriori estimation of Markov Random Fields and are suitable to solve accurately many different problems in image analysis, including image segmentation. In this paper we present a two-stage graph cut based model for segmentation of touching cell nuclei in fluorescence microscopy images. In the first stage voxels with very high probability of being foreground or background are found and separated by a boundary with a minimal geodesic length. In the second stage the obtained clusters are split into isolated cells by combining image gradient information and incorporated a priori knowledge about the shape of the nuclei. Moreover, these two qualities can be easily balanced using a single user parameter. Preliminary tests on real data show promising results of the method.

1 Introduction

Image segmentation is one of the most crucial tasks in fluorescence microscopy and image cytometry. Due to its importance many methods were proposed for solving this problem in the past. For simple cases basic techniques like thresholding [1], region growing [2] or watershed algorithm [2] are the most popular. However, when the data is severely degraded or contains complex structures requiring isolation of touching objects these simple methods are not powerful enough. Unfortunately, these scenarios are quite frequent. For this type of images more sophisticated methods have been designed in the past [3,4,5]. Their results although quite satisfactory, have some limitations: 1) in some cases suffer from over/undersegmentation, 2) need for human input, 3) require specific preparation of the biological samples.

The graph cut segmentation framework, first outlined by Boykov and Jolly [6,7], received a lot of attention in the recent years due to its robustness, reasonable computational demands and the ability to integrate visual cues, contextual information and topological constraints while offering several favourable characteristics

A.-B. Salberg, J.Y. Hardeberg, and R. Jenssen (Eds.): SCIA 2009, LNCS 5575, pp. 410–419, 2009.

like global optima [8], unrestricted topological properties and applicability to N-D problems. The core of their solution relies on modeling the segmentation process as a labelling problem with an associated energy function. This function is then optimized by finding a minimal cut in a specially designed graph. The method can be also formulated in terms of Maximum a Posteriori estimate of a Markov Random Field (MAP-MRF) [9,10].

In this paper we present a two-stage fully automated graph cut based model for segmentation of touching cell nuclei addressing most of the problems associated with the segmentation of fluorescence microscopy images. In the first stage background segmentation is performed. Voxels with very high probability of being foreground or background are located and separated by a boundary with a minimal geodesic length. In the second stage the obtained clusters are split into isolated cells by combining image gradient information and incorporated a priori knowledge about the shape of the nuclei. Moreover, these two qualities can be easily balanced using a single user parameter, allowing to control the placement of the dividing line in a desired way. This is a great advantage over the standard methods. Our algorithm can work on both 2-D and 3-D data sets. We demonstrate its potential on segmentation of 2-D cancer cell line images.

The organization of the paper is as follows. Graph cut segmentation framework is briefly reviewed in Section 2. A detailed description of our two-stage model is presented in Section 3 with experimental results in Section 4. In Section 5 we discuss the benefits and limitations of our method. Finally, we conclude our work in Section 6.

2 Graph Cut Segmentation Framework

In this section we briefly revisit the graph cut segmentation framework and related terms [6,7,11,10]. Because our method exploits both two-terminal and multi-terminal graph cuts we are going to describe the latter case which is a generalization of the former.

Consider an N-D image $\mathcal{I}$ consisting of set of voxels $\mathcal{P}$ and some neighbourhood system, denoted $\mathcal{N}$, containing all unordered pairs $\{p, q\}$ of neighbouring elements in $\mathcal{P}$. Further, let us consider a set of labels $\mathcal{L} = \{l_1, l_2, \ldots, l_n\}$ that should be assigned to each voxel in the image. Now, let $A = (A_1, \ldots, A_{|\mathcal{P}|})$ be a vector, where $A_i \in \{1, \ldots, n\}$ specifies the assignment of labels $\mathcal{L}$ to voxels $\mathcal{P}$.

The energy corresponding to a given labelling A is constructed as a linear combination of regional (data dependent) and boundary (smoothness) term and takes the form of:

$$E(A) = \lambda \cdot \sum_{p \in \mathcal{P}} R_p(A_p) + \sum_{(p,q) \in \mathcal{N}} B_{(p,q)} \cdot \delta_{A_p \neq A_q}, \tag{1}$$

where $R_p(l)$ is the regional term evaluating the penalty for assigning voxel p to label l, $B_{(p,q)}$ is the boundary term evaluating the penalty for assigning neighbouring voxels p and q to different labels, δ is the Kronecker delta and λ is a weighting factor. The choice of the two evaluating functions R_p and $B_{(p,q)}$ is

crucial for the segmentation. Based on the underlying MAP-MRF, the values of R_p are usually calculated as follows:

$$R_p(l) = -\log \Pr(p|l), \tag{2}$$

where $\Pr(p|l)$ is the probability that voxel p matches the label l. It is assumed that these probabilities are known a priori. However, in practice it is often hard to estimate them. The boundary term function can be naturaly expressed using the image contrast information [6,7] and can also approximate any Euclidean or Riemmannian metric [12]. The choice of $B_{(p,q)}$ for cell nuclei segmentation is discussed in Sect. 3.1.

Equation 1 can be minimized by finding a minimal cut in a specially designed graph (network). Construction of such graph is depicted in Fig. 1. In the first step a node is added for each voxel and these nodes are connected according to the neighbourhood $\mathcal{N}$. The edges connecting these nodes are denoted n-links and their weights (capacities) are determined by the function $B_{(p,q)}$. In the next step terminal nodes $\{t_1, t_2, \ldots, t_n\}$ corresponding to labels in $\mathcal{L}$ are added and each of them is connected with all nodes created in the first step. The resulting edges are called t-links and their capacities are given by the function R_p [10].

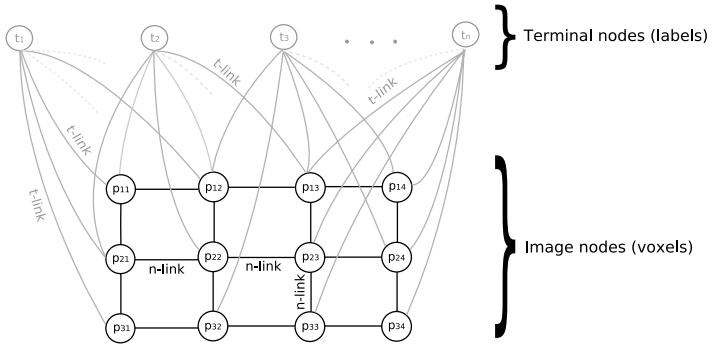

Fig. 1. Graph construction for given 2-D image, $\mathcal{N}_4$ neighbourhood system and set of terminals $\{t_1, \ldots, t_n\}$ (not all t-links are included for the sake of lucidity)

The minimal cut splits the graph into disjoint components $C_1, \ldots, C_n$, such that t_i lies in C_i for all $i \in \{1, \ldots, n\}$ and the sum of capacities of the removed edges is minimal. Consequently, every voxel receives the label of the terminal node in its component. In case of only two labels (terminals) the minimal cut can be found effectively in polynomial time using one of the well-known max-flow algorithms [11]. Unfortunately, for more than two terminals the problem is NP-complete [13] and an approximation of the minimal cut is calculated [10]. In this framework it is also possible to set up hard constraints in an elegant way. A binding of voxel p to a chosen label $\hat{l}$ is done by setting $R_p(l \neq \hat{l}) = \infty$ (refer to [7] for implementation details).

3 Cell Nuclei Segmentation

In this section we are going to give a detailed description of our fully automated two-stage graph cut model for segmentation of touching cell nuclei. The images that we cope with are acquired using fluorescence microscopy, meaning they are blurred, noisy and low contrast. They contain bright objects of mostly spherical shape on a dark background. Also the nuclei are often tightly packed and form clusters with indistinctive frontiers. Moreover, the interior of the nuclei can be greatly non-homogeneous and can contain dark holes incised into the nucleus boundary (caused by nucleoli, non-uniformity of chromatin organization or imperfect staining). See Sect. 4 for examples of such data.

In the first stage of our method foreground/background segmentation is performed, while in the second stage individual cells are identified in the obtained cell clusters and separated. The algorithm can work on both 2-D and 3-D data sets.

3.1 Background Segmentation

In this stage we are interested in binary labelling of the voxels with either a foreground or background label. The voxels that receive the foreground label are then treated as cluster masks and are separated into individual nuclei in the second stage. Because we deal with binary labelling only, the standard two-terminal graph cut algorithm [7] together with fast optimization methods [11] can be used. To obtain correct segmentation of the background, functions $B_{(p,q)}$ and R_p in (1) have to be set properly.

As the choice for $B_{(p,q)}$ we suggest the Riemmanian metric based edge capacities proposed in [12]. The equations in [12] can be simplified to the following form (assumming p and q are voxel coordinates):

$$B_{(p,q)} = \frac{\|q-p\|^2 \cdot \Delta\Phi \cdot g(p)}{2 \cdot \left[g(p) \cdot \|q-p\|^2 + (1-g(p)) \cdot \left\langle q-p, \frac{\nabla I_p}{|\nabla I_p|} \right\rangle^2 \right]^{\frac{3}{2}}}, \tag{3}$$

where $\Delta\Phi$ is $\frac{\pi}{4}$ for 8-neighbourhood and $\frac{\pi}{2}$ for 4-neighbourhood system respectively, $\langle\cdot\rangle$ denotes the dot product, ∇I_p is image gradient in voxel p and

$$g(p) = \exp\left(-\frac{|\nabla I_p|^2}{2\sigma^2}\right), \tag{4}$$

with σ being estimated as the average gradient magnitude in the image. Note that this equation applies to the 2-D case and that it is slightly different for 3-D [12]. It is also advisable to smooth the input image (e.g. using a Gaussian filter) before calculating the capacities.

Setting the capacities of t-links is the tricky part of this stage. In most approaches [5] homogeneous interior of the nuclei is assumed, allowing some simplifications of the algorithms. While this may be true in some situations, often it

is not, as mentioned before. Hence, it is really hard to estimate the probability of the voxel being foreground or background based solely on its intensity. For example, the bright voxels among the cell nuclei in the top cluster in Fig. 2 are part of the background. To avoid introduction of false information into the model we suggest to stick to hard constraints only. We place them into voxels with very high probability of being background or foreground and ignore the intensity information elsewhere.[1] To find such voxels in the image we perform bilevel histogram analysis, find the two peaks corresponding to background and foreground and take the centres of these two peaks as our background/foreground thresholds. For voxels with intensity below the background threshold (black pixels in Fig. 2b) the corresponding capacity of the t-link going to background terminal is set to ∞ and analogously for voxels with intensity above the foreground threshold (white pixels in Fig. 2b). Remaining voxels (grey pixels in Fig. 2b) are left without any affiliation and both their t-link capacities are set to zero. As a consequence, λ value in (1) is irrelevant in this situation.

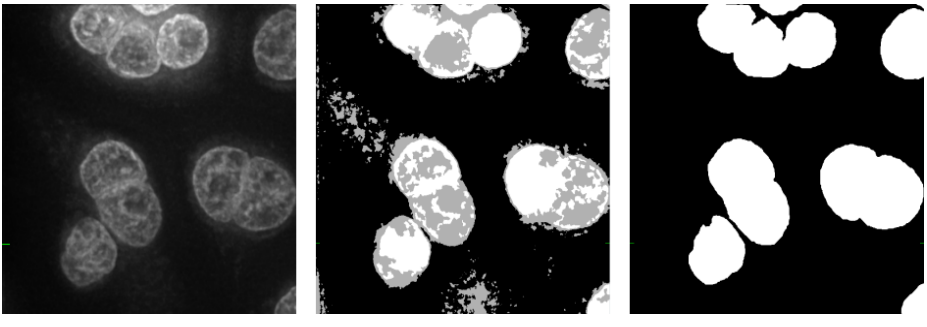

Fig. 2. Background segmentation. (a) Original image. (b) Foreground (white) and background (black) markers (preprocessing mentioned in Sect. 4 was used). (c) Background segmentation.

Finally, finding the minimal cut in the corresponding network while using the capacities described in this subsection gives us the background segmentation, that is shown in Fig. 2c. The result is a segmentation separating the background and foreground hard constraints with a minimal geodesic boundary length with respect to chosen metric. It is worth mentioning, that due to the nature of graph cuts, effective interactive correction of the segmentation could be involved at this stage of the method whenever required.

3.2 Cluster Separation

Whereas in the first stage of our method the segmentation is driven largely by the image gradient (n-links), trying to satisfy the hard constraints at the same

[1] Note that the intensity gradient information is included in n-link weights.

time, in the second stage we employ a different approach and stick to the cluster morphology. That is motivated by the fact, that the image gradient inside of the nuclei does not provide us with reliable information. The interior of the nuclei can be greatly non-homogeneous and the dividing line between the touching nuclei not distinct enough, while some other parts of the nuclei can contain very sharp gradients. However, our solution allows us to tune the algorithm to different scenarios by simply changing the value of the parameter λ in (1). The clusters obtained in the first stage are treated separately in the second stage, so the following procedures refer to the process of division of one particular cluster.

First of all, the number of cell nuclei in the cluster is estabilished. To do this we calculate a distance transform of the cluster interior and find peaks in the resulting image using a morphological extended maxima transformation [2] with the maxima height chosen as 5% of the maximum value. The number of peaks in the distance transform is then taken as the number of cell nuclei in the cluster. If the cluster contains only one cell nucleus the second stage is over, otherwise we proceed to the separation of the touching nuclei. In the following text we will denote M_l the connected set of voxels corresponding to one peak in the distance transform, where $l \in \{1, \ldots, n\}$ and n is the number of nuclei in the cluster. An estimation of the nucleus radius σ_l is calculated as the mean value of the distance transform across voxels in M_l for each nucleus.

To find the dividing line among the cell nuclei a graph cut in a network with n terminals is used. The n-link capacities are set up in exactly the same way as in the first stage. The t-link weights are assigned as follows. For each label l and each voxel p in the cluster mask we define $d_l(p)$ to be the Euclidean distance of the voxel p to the nearest voxel in M_l. The values of d_l for all voxels and labels can be effectively calculated using n distance transforms. Further, we estimate the probability of voxel p matching label l as:

$$\Pr(p|l) = \exp\left(-\frac{d_l(p)^2}{2\sigma_l}\right), \tag{5}$$

which corresponds to a normal distribution with the probability inversely proportional to the distance of the voxel p from the set M_l and standard deviation $\sqrt{\sigma_l}$. The normalizing factor is omitted to ensure uniform amplitude of the probabilities. As a consequence of (2) the regional penalties are calculated as:

$$R_p(l) = -\log \Pr(p|l) = \frac{d_l(p)^2}{2\sigma_l} \quad . \tag{6}$$

Indeed, hard constraints are set up for voxels in M_l. Such regional penalties (proportional to the distance from the M_l sets) incorporate an a priori shape information into the model and help us to push the dividing line of the neighbouring nuclei to its expected position and ignore the possibly strong gradients near the nucleus center. How much it will be pushed depends on the parameter λ in (1). The influence of this parameter is illustrated in Fig. 3. Generally, the smaller λ is, the higher importance will be given to the image gradient.

If the given cluster contains more than two cell nuclei (and more than two terminals in consequence) standard max-flow algorithms can not be used to find

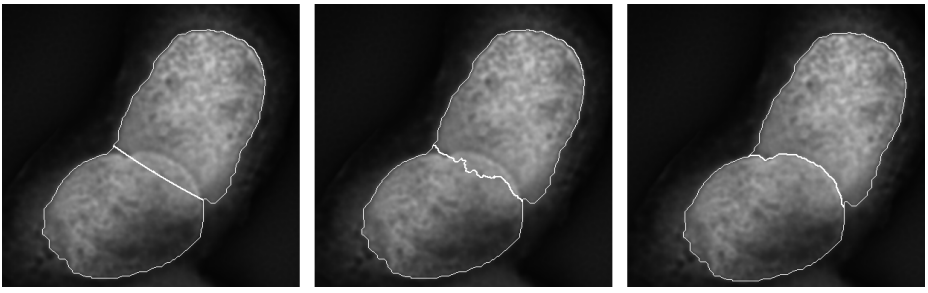

Fig. 3. Influence of the λ parameter on data with distinct frontier between the nuclei. (a) $\lambda = 1000$ (b) $\lambda = 0.15$ (c) $\lambda = 0$.

the minimal cut. Due to the NP-completeness of the problem [13], it is necessary to use approximations. We use the α-β-swap iterative algorithm proposed in [10], that is based on repeated calculations of standard minimal cut for all pairs of labels.[2] According to our tests this approximation converges very fast and three or four iterations are usually enough to reach the minimum. To obtain an initial labelling we assign a label l' to voxel p such as $l' = \arg\min_{l \in \mathcal{L}} R_p(l)$.

4 Experimental Results

Results obtained using an implementation of our model for 2-D images are presented in this section. We have tested our method on two different data sets. The first one consisted of 40 images (16-bit grayscale, 1300×1030 pixels) of DAPI stained HL60 (human promyelocytic leukemia cells) cell nuclei. The second one consisted of 10 images (16-bit grayscale, 1392×1040 pixels) of DAPI stained A549 (lung epithelial cells) cell nuclei deconvolved using the Maximum Likelihood Estimation algorithm, provided by the Huygens software (Scientific Volume Imaging BV, Hilversum, The Netherlands). In both cases the 2-D images were obtained as maximum intensity projections of 3-D images to the xy plane. Samples of the final segmentation are depicted in Fig. 4.

Each of the images in the data sets consisted of 10 to 20 clustered cell nuclei. Even though the clusters are quite complicated (particularly in the HL60 case) and the image quality is low, all of the nuclei are reliably identified, as can be seen in the figure. To quantitatively measure the accuracy of the segmentation, we have used the following sensitivity and specificity measures with respect to an expert provided ground truth:

$$Sens_i(f) = \frac{TP_i}{TP_i + FN_i} \qquad Spec_i(f) = \frac{TN_i}{TN_i + FP_i}, \tag{7}$$

[2] It is also possible to use the stronger α-expansion algorithm described in the same paper, because our $B_{(p,q)}$ is a metric.

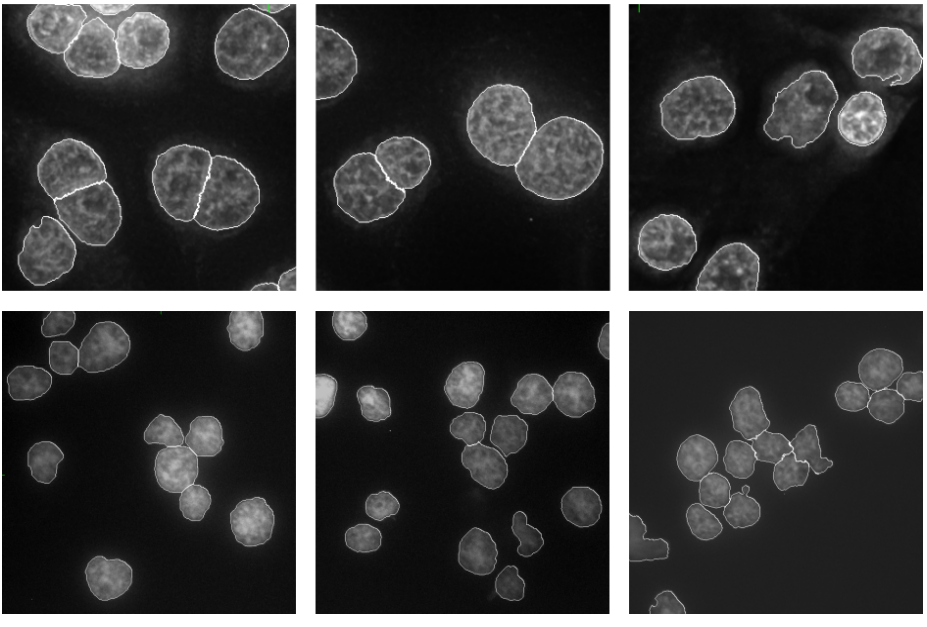

Fig. 4. Samples of the final segmentation. Top row: A549 cell nuclei. Bottom row: HL60 cell nuclei.

where i is a particular cell nucleus, f is the final segmentation and TP_i (true positive), TN_i (true negative), FP_i (false positive) and FN_i (false negative) denote the number of voxels correctly (true) and incorrectly (false) segmented as nucleus i (positive) and background or another nucleus (negative), respectively. Average and worst case values of both measures are listed in Table 1.

Table 1. Quantitative evaluation of the segmentation. Average and worst case values of *sensitivity* and *specificity* measures calculated against expert provided ground truth.

Cell line	$Sens_{worst}(f)$	$Spec_{worst}(f)$	$Sens_{avg}(f)$	$Spec_{avg}(f)$
A549	91.42%	92.98%	98.38%	97.00%
HL60	88.60%	95.68%	97.43%	98.12%

The computational time demands and memory consumption of our algorithm are listed in Table 2, they were approximately the same for both data sets (measured on a PC equipped with an Intel Q6600 processor and 2 GB RAM). The standard max-flow algorithm [7] was used to find the minimal cut in two-terminal networks. The memory footprint is smaller in the second stage, that is due to the fact that only parts of the image are processed. Also the computational time of the second stage depends on the number of nuclei clusters and on their complexity.

Table 2. Computational demands on tested images ($\approx 1300 \times 1000$ pixels)

Stage	Total time	Peak memory consumption
1	2 sec	150 MB
2	5 sec	30 MB

For the segmentation of HL60 cell nuclei $\lambda = 0.001$ was used, because the interior of the nuclei is quite homegeneous and the dividing lines are perceptible. In the second case, $\lambda = 0.15$ was used, giving lower weight to the gradient information. Image preprocessing consisted of smoothing and background illumination correction in the first case and white top hat transformation followed by a morphological hole filling algorithm [2] in the second.

5 Discussion

The method described in this paper is fully automatic with the only tunable parameter being the λ weighting factor. For higher values of λ the segmentation is driven mostly by the regional term incorporating the a priori shape knowledge, for lower by the image gradient. In some cases (data with distinct frontier between the nuclei, such as the one in Fig. 3) it is even possible to use $\lambda = 0$. Such simple tuning of the algorithm is not possible with standard methods.

An important aspect of the second stage of our method is the incorporation of a priori shape information into the model. The proposed approach is well suited to a wide range of shapes, not only circular, provided that the M_l sets mentioned in Sect. 3.2 approximate the skeletons of the objects being sought. It is obvious that in case of mostly circular nuclei the skeletons correspond to centres and our method looking for peaks in the distance transform of the cluster is applicable. However, in case of more complex shapes it might be harder to find the initial M_l sets and the number of objects.

The implementation of our method in 3-D is straightforward. However, some complications may arise, which include a slower computation due to the huge size of the graphs and those related to low resolution and significant blur of the fluorescence microscope images in the axial direction.

6 Conclusion

A fully automated two-stage segmentation method based on the graph cut framework for the segmentation of touching cell nuclei in fluorescence microscopy has been presented in this paper. Our main contribution was to show how to cope with low image quality that is unfortunately common in optical microscopy. This is achieved particularly by combining image gradient information and incorporated a priori knowledge about the shape of the nuclei. Moreover, these two qualities can be easily balanced using a single user parameter.

We plan to compare the proposed approach with other segmentation methods, in particular, level-sets and the watershed transform. The quantitative evaluation

in terms of computational time and accuracy will be done on both synthetic data with a ground truth and real images. Our goal is also to implement the method in 3-D and improve its robustness for more complex types of clusters, that appear in thick tissue sections.

Acknowledgments. This work has been supported by the Ministry of Education of the Czech Republic (Projects No. MSM-0021622419, No. LC535 and No. 2B06052). COS and AMB were supported by the Marie Curie IRG Program (grant number MIRG CT-2005-028342), and by the Spanish Ministry of Science and Education, under grant MCYT TEC 2005-04732 and the Ramon y Cajal Fellowship Program.

References

1. Pratt, W.K.: Digital Image Processing. Wiley, Chichester (1991)
2. Soille, P.: Morphological Image Analysis, 2nd edn. Springer, Heidelberg (2004)
3. Ortiz de Solórzano, C., Malladi, R., Leliévre, S.A., Lockett, S.J.: Segmentation of nuclei and cells using membrane related protein markers. Journal of Microscopy 201, 404–415 (2001)
4. Malpica, N., Ortiz de Solórzano, C., Vaquero, J.J., Santos, A., Lockett, S.J., Vallcorba, I., García-Sagredo, J.M., Pozo, F.d.: Applying watershed algorithms to the segmentation of clustered nuclei. Cytometry 28, 289–297 (1997)
5. Nilsson, B., Heyden, A.: Segmentation of dense leukocyte clusters. In: Proceedings of the IEEE Workshop on Mathematical Methods in Biomedical Image Analysis, pp. 221–227 (2001)
6. Boykov, Y., Jolly, M.P.: Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images. In: IEEE International Conference on Computer Vision, July 2001, vol. 1, pp. 105–112 (2001)
7. Boykov, Y., Funka-Lea, G.: Graph cuts and efficient n-d image segmentation. International Journal of Computer Vision 70(2), 109–131 (2006)
8. Kolmogorov, V., Zabih, R.: What energy functions can be minimized via graph cuts? IEEE Transactions on Pattern Analysis and Machine Intelligence 26(2), 147–159 (2004)
9. Boykov, Y., Veksler, O., Zabih, R.: Markov random fields with efficient approximations. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 648–655. IEEE Computer Society, Los Alamitos (1998)
10. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. IEEE Transactions on Pattern Analysis and Machine Intelligence 23, 1222–1239 (2001)
11. Boykov, Y., Kolmogorov, V.: An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. IEEE Transactions on Pattern Analysis and Machine Intelligence 26(9), 1124–1137 (2004)
12. Boykov, Y., Kolmogorov, V.: Computing geodesics and minimal surfaces via graph cuts. In: IEEE International Conference on Computer Vision, vol. 1, pp. 26–33 (2003)
13. Dahlhaus, E., Johnson, D.S., Papadimitriou, C.H., Seymour, P.D., Yannakakis, M.: The complexity of multiterminal cuts. SIAM J. Comput. 23(4), 864–894 (1994)