7

Computational Intelligence Techniques in Image Segmentation for Cytopathology

Andrzej Obuchowicz, Maciej Hrebień, Tomasz Nieczkowski,
and Andrzej Marciniak

Institute of Control and Computation Engineering
University of Zielona Góra
ul. Podgórna 50, 65-246 Zielona Góra, Poland
{`A.Obuchowicz,M.Hrebien,T.Nieczkowski,A.Marciniak`}`@issi.uz.zgora.pl`

Summary. A variety of computational intelligence approaches to nuclei segmentation in the microscope images of fine needle biopsy material is presented in this chapter. The segmentation is one of the most important steps of the automatic medical diagnosis based on the analysis of the microscopic images, and is crucial to making a correct diagnostic decision. Due to complex nature of biological images, standard segmentation methods are not effective enough. In this chapter we present and discuss some modified versions of watershed algorithm, active contours, cellular automata, Grow-Cut technique, as well as new approaches like fuzzy sets of I and II type, and the sonar-like method.

7.1 Introduction

Segmentation of the object of interest is one of the most critical tasks in image analysis and therefore it has been the subject of considerable research activity over the last four decades. During all this time we have witnessed a tremendous development of new, powerful instruments for detecting, storing, transmitting, and displaying images but automatic segmentation still remained a challenging problem.

This fact is easy to notice in medical applications, where image segmentation is particularly difficult due to restrictions imposed by image acquisition, pathology and biological variation. Biomedical image segmentation is a sufficiently complex problem that no single strategy has proven to be completely effective. Due to a complex nature of biomedical images, it is practically impossible to select or develop automatic segmentation methods of generic nature, that could be applied for any type of these images, e.g. for either micro- and macroscopic images, cytological and histological ones, MRI and X-ray, and so on.

In this chapter we are focused on the microscopic images of the Fine Needle Biopsy (FNB) material taken from the breast cancer. In the last decade we have been observing a dynamic growth in the number of research works conducted in the area of breast cancer diagnosis. Many university centers and commercial institutions [18] are focused on this issue because of the fact that breast cancer is becoming the most common form of cancer disease of today's female population. The attention covers not only curing the external effects of the disease [2, 43]

T.G. Smolinski et al. (Eds.): Comp. Intel. in Biomed. & Bioinform., SCI 151, pp. 169–199, 2008.
springerlink.com

but also its fast detection in the early stadium. Thus, the construction of a fully automatic cancer diagnosis system supporting a human expert has became a challenging task.

Many nowadays camera-based automatic breast cancer diagnosis systems have to face the problem of cells and their nuclei separation form the rest of the image content [19, 35, 38, 48]. This process is very important because the nucleus of the cell is the place where breast cancer malignancy can be observed. Thus, much attention in the construction of the expert supporting diagnosis system have to be paid to the segmentation stage.

The main difficulty of the segmentation process is due to incompleteness and uncertainty of the information contained in the image. Imperfection of the data acquisition process in the form of noise, chromatic distortion and deformity of the cytological material caused by its preparation additionally increases the complicity of the problem. The nature of the image acquisition (3D to 2D transformation) and the method of scene illumination also affects the image's luminance and sharpness. In many cases one must also deal with a low-cost CCD sensor whose quality and resolution capabilities are rather small.

Until now many segmentation methods have been proposed [3, 4, 16, 33, 47, 42] but, unfortunately, each of them introduces different kinds of additional problems and usually works in practice under given assumptions and/or needs the end-user's interaction/co-operation [19, 41, 48, 49]. Since many nowadays cytological projects assume full automation and real-time operation with a high degree of efficacy, a method free of drawbacks of the already known approaches has to be constructed.

The aim of this chapter is presentation a variety of computational intelligence approaches to nuclei segmentation in the microscope images of fine needle biopsy material. Some of them are modified versions of cytological image segmentation methods adopted for fine needle biopsy images, that is the watershed algorithm [10], active contours [11], cellular automata [27], GrowCut technique [9], and decision three technique [29]. Some of them exemplify quite new proposals of segmentation techniques: sets of I and II type approach [7], the sonar-like method [28]. One can also find here a description of the denoising and contrast enhancement techniques, pre-segmentation and a fully automatic nuclei localization mechanism used in our approaches. The quality and applicability of described segmentation methods are still investigated by our team. The final judgement can be precised after finishing the next steps of automatical breast cancer diagnosis, i.e. morphometric parameters calculating and classification. The quality of classification will be testimony of the applied segmentation.

7.2 Image Segmentation of Cytological FNB Microscope Images

7.2.1 Segmentation within Image Analysis

Image segmentation consists in subdividing an image into its constituent regions that hopefully correspond to structural units in the scene or distinguish objects

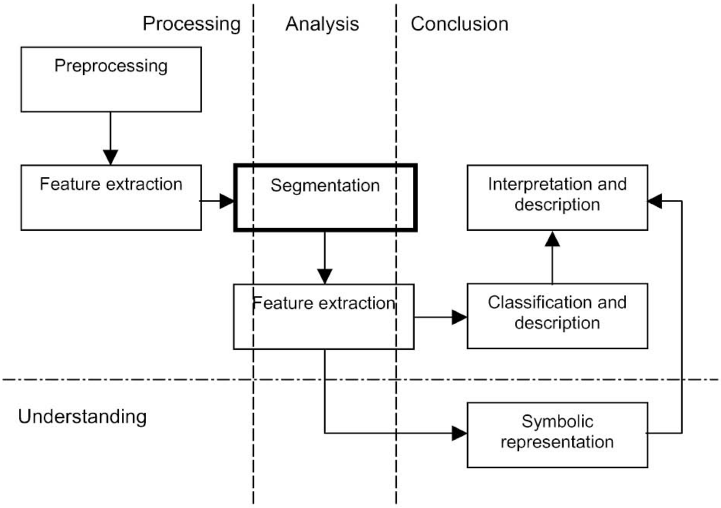


Fig. 7.1. An automatic vision system

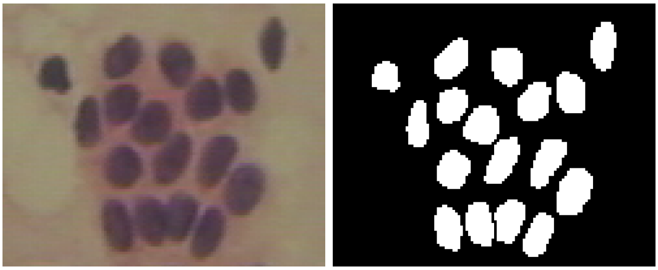
of interest. The level to which this subdivision is carried depends on the problem being solved and applied approach. In the Fig.7.1 one can notice that segmentation is a crucial and a central step in whole image analysis system. The dashed lines in the picture illustrate vagueness of the definitions adopted in literature (see e.g. [26, 36]). Some of them are deliberately restrictive and assume that no contextual information is utilized in the segmentation [36]. In this approach segmentation does not involve identifying and isolating segments. The process consists in subdividing an image according to some homogeneity criteria defined for individual pixels; it does not attempt to recognize the individual segments nor their relationships to one another. In the most opposite approach, segmentation is defined as a process of isolation of components that correspond to the physical objects in the scene. In this case, a feedback from the subsequent steps is taken into account, i.e. analysis outcomes and relations between isolated image regions are important factors constituting the segmentation criteria. In life-crucial applications (e.g. medical diagnosis) even the intervention of a human operator is required. Different patterns of such interaction can be found in [14, 26, 32].

As a consequence of application-driven approach in defining the process of segmentation, no single standard methodology has emerged. In principle, there is no general theory of image segmentation, though some attempts to build a functional model have been made (see for example [50]). Moreover, the problem of segmentation is ill-defined and can be perceived as one of psychophysical perception and therefore not susceptible to a purely analytical solution [8]. As a consequence, no single standard method of image segmentation has been elaborated, although there are methods that have received some degree of popularity.

7.2.2 Testing Database of Images

What we have on input is the cytological images database of the Fine Needle Biopsy material gained in a cooperation with experts from the Zielona Góra's *Onkomed* medical center [21, 22]. The database consists of 750 images of 75 clinical cases, including 25 images of benign tumour, 25 of malignant tumour and 25 of fibroadenoma. There are 10 images for every case - one of them is magnified by 100 and 9 magnified by 400. Photographs x400 where selected in such a way as to contain at least 10 nucleus suitable for the morphometric analysis. As a result, the set of images for one case contains at least 90 nucleuses to analyze. The image itself is coded using the RGB colorspace and is not subject to any kind of lossy compression (a raw color bitmap format), and with a resolution of 704 × 578 pixels and 24-bit color depth (16.7M colors). The number of distinct colors in images varies from about 10 to 60 thousands.

What we expect on output is a binary segmentation mask with one pixel separation rule which will allow more robust morphometric parameters estimation in our future work. Additionally, the proposed segmentation algorithm should be insensitive to colors of contrasting pigments used for preparation of the cytological material (see an example in Fig. 7.2).



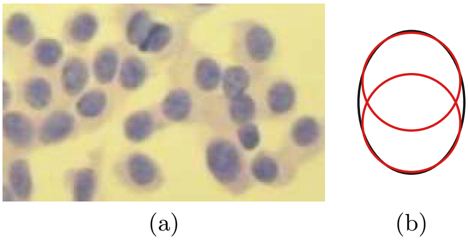
(a) (b)

Fig. 7.2. Exemplary fragment of: (a) cytological image, (b) appropriate segmentation mask

7.2.3 Image Filtering and Preparation

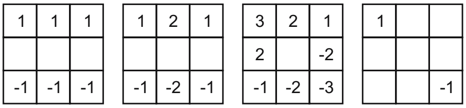
The quantity of information contained in a color image is surplus at the early stage of image processing. The color components do not carry as important information as luminosity so they can be removed to reduce processing complexity. An RGB color image can be converted to greyscale by calculating a luminance value in the same way as it is calculated for YCbCr color space [37].

Since a great deal of images have a low contrast, an enhancement technique is needed to improve theirs quality. In our research we use a simple histogram processing with the linear transform of images levels of intensities, namely a cumulated sum approach [36].



(a) (b)

Fig. 7.3. Examplary fragment of cytological image with circular nuclei



1	1	1
-1	-1	-1

1	2	1
-1	-2	-1

3	2	1
2		-2
-1	-2	-3

1		
		-1

Fig. 7.4. Gradient masks used in our experiments

If we look closely at the nuclei we have to segment, we notice that they all have an elliptical shape (Fig. 7.3). Most of them resemble an ellipse but, unfortunately, detection of the ellipse is computationally expensive. The shape of the ellipse can be approximated by a given number of circles (as shown in Fig. 7.3b). The detection of circles is much simpler in the sense of the required computations because we have only one parameter, which is the radius R. These observations and simplifications constitute grounding for a nucleus pre-segmentation algorithm – in our approach we try to find such circles with different radii in a given feature space.

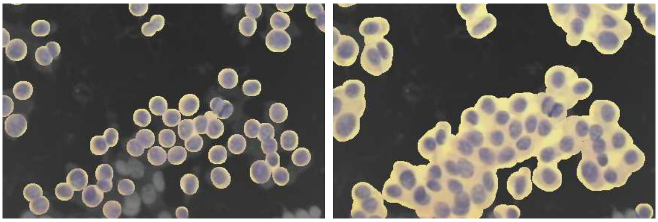
The Hough transform [1, 6, 51] can be easily adopted for the purpose of circle detection. The transform in a discrete space can be defined by:

$$HT_{discr}(R, i_0, j_0) = \sum_{i=i_0-R}^{i_0+R} \sum_{j=j_0-R}^{j_0+R} g(i,j)\delta\Big((i-i_0)^2 + (j-j_0)^2 - R^2\Big), \quad (7.1)$$

where g is a two-dimensional feature image and δ is Kronecker's delta (equal to unity at zero). HT_{discr} plays the role of an accumulator which accumulates the levels of feature image g similarity to the circle placed at the (i_0, j_0) position and defined by the radius R.

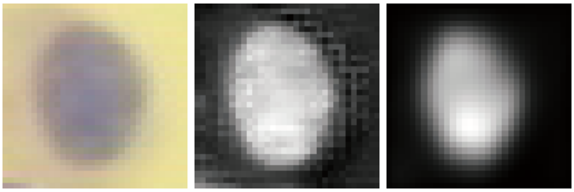
The feature space g can be created by many different ways. In our approach we use a gradient image as the feature indicating the occurrence or absence of the nucleus in a given fragment of the cytological image. The gradient image is a saturated sum of gradients estimated in eight directions on the greyscale image prepared in the pre-processing stage. The base gradients can be calculated using, e.g., Prewitt's, Sobel's mask methods [12, 44] or their heavy or light versions (Fig. 7.4).

Thresholding the values in the accumulator by a given θ value we can obtain a very good pre-segmentation mechanism with a lower threshold strategy (see, for



(a) (b)

Fig. 7.5. Exemplary results of the pre-segmentation stage for two different θ threshold strategies: (a) high and (b) low



(a) (b) (c)

Fig. 7.6. Exemplary fragment of: (a) cytological image, (b) Euclidian distance to the mean background color, (c) smoothed out version of (b)

instance, Fig. 7.5). Since the threshold value strongly depends on the database and the feature image g, the method can only be used as a pre-segmentation stage. A smaller value of the threshold causes fast removal of unimportant information from the background, and what we achieve is a mask, which approximately defines the places where are the objects we have to segment (nuclei in this case) and where is the background. Such a mask can constitute a base for more sophisticated and detail-oriented algorithms.

The results obtained from the pre-segmentation stage can lead us to the estimation of an average background color. Such information can be used to model the nuclei as a color distance between the background and the objects, which fulfils the requirements of the lack of any color dependency in the imaged material (the color of contrasting pigments may change in the future). In our research we tried few distance metrics: Manhattan's, Chebyshev's, the absolute Hue value from the HSV colorspace, but the Euclidian one gives us visually the best results (Fig. 7.6ab):

$$D_{euclid} = \sqrt{(I_R - B_R)^2 + (I_G - B_G)^2 + (I_B - B_B)^2}, \qquad (7.2)$$

where B is the average background color estimated for the I input image.

Since the modeling distance can vary in the local neighborhood (see Fig. 7.6b), mostly because of camera sensor simplifications, a smoothing technique is needed to reconstruct the nuclei shape. The smoothing operation in our approach relies on the fact that this sort of 2D signal can be modeled as a sum of sinusoids [20] with defined amplitudes, phase shifts and frequencies. Cutting all low amplitude frequencies off (leaving only a few significant ones with the highest amplitude) will result in a signal deprived of our problematic local noise effect (Fig. 7.6c). What we finally achieve is a three-dimensional modeled terrain where hills correspond to nuclei.

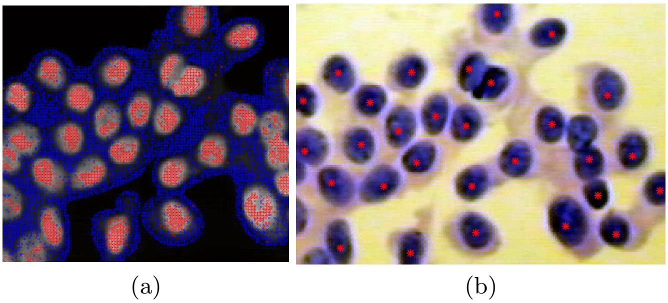
The localization of objects on a modeled map of nuclei can be performed locally using various methods. In our approach we have chosen an evolutionary (1+1) search strategy [25, 30, 31] mostly because it is simple, quite fast despite appearances, can be easily parallelized due to its nature and settles very well in local extrema, which is very important in our case.

The search in our approach can be conducted in two versions: single-point and multi-point. In the single-point version it is only allowed to have only one marker pointing a nucleus while in the multi-point one it is allowed to have more than one marker pointing the same nucleus.

The used watershed algorithm as a final segmentation method forced us to create two population of individuals. The first population localizes the background. Specimens are moved through the mutation stage $Y_i^t = X_i^t + r^t N_i(0,1)$ with a constant movement step ($r^t = 1$) preferably to places with a smaller density of population to maximize background coverage. The second population localizes the nuclei. Specimens are moved with a decreasing movement step ($r^t = R_{max}(1/R_{max})^{t/t_{max}}$) to group very fast the population near local extremum in the first few epochs and to finally work on details in the ending ones. The movement of individuals is preferred to be directed towards places with a higher population density to create the effect of nuclei localization.

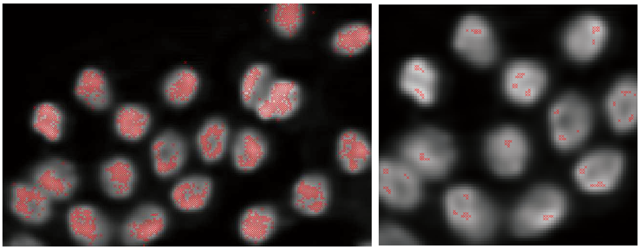
The fitness function ϕ calculates the average *height* of the terrain in a given position including the nearest neighborhood defined by the smallest radius detected by the Hough transform in the pre-segmentation stage. Such a definition of the fitness function avoids a possible split of the population, localized near a nucleus with multimodal character of its shape, giving only one marker for a nucleus (Fig. 7.7b).

Finally, the nucleus is localized in the place where the density of the population searching for hilltops in the modeled terrain is locally maximal. As we have



(a) (b)

Fig. 7.7. Exemplary single-point localization: (a) screenshot after 8 epochs, (b) final result (localization points are marked with red asterisks)



(a) (b)

Fig. 7.8. Exemplary multi-point localization: (a) screenshot after few epochs, (b) final result

mentioned earlier, the method is quite fast and just a few epochs are needed to observe a visible progress in nuclei localization and background coverage (Fig. 7.7a).

The algorithms that do not have such tight requirements concerning only one single marker per nucleus, that is they allow multiple markers pointing the same one, not optimal or even false localization points and can take information about the background location from the pre-segmentation mask [9], can use much simplified version of the above presented $(1 + 1)$ search strategy. In such cases we can use only one population, that is the one searching for nuclei and the fitness function is simply the terrain *height* at an individuals position. The number of iterations of the algorithm can also be reduced, because we need only an approximate localization of nuclei (Fig. 7.8). Thus, the algorithm is the same and the only difference with the one described above is the fitness function ϕ and reduced number of epochs.

7.2.4 Watershed Segmentation

Method description

The watershed segmentation algorithm is inspired by natural observations, such as a rainy day in the mountains [12, 36, 37]. A given image can be defined as a terrain on which nuclei correspond to valleys (upside down the terrain modeled in previous steps). The terrain is flooded by rainwater and arising puddles are starting to turn into basins. When the water from one basin begins to pour away to another, a separating watershed is created.

The flooding operation has to be stopped when the water level reaches a given θ threshold. The threshold should preferably be placed somewhere in the middle between the background and a nucleus localization point. In our approach the nuclei are flooded to the half of the altitude between the nucleus localization point and the average height of the background in the local neighborhood. Since the images we have to deal with are spot illuminated during the imaging

```
// For each basin
$\forall p \in P$ assign a label $(i + 1)$

for $\theta \in 0 : \Delta : 1$
    for $\forall p \in P$
        color $p$ to the level $\leqslant \theta * \frac{\Psi(p)}{2}$
    end
end
```

Algorithm 1. The simplified version of the watershed algorithm

operation (resulting in a modeled terrain higher in the center of the image and much lower in the corners), this mechanism protects the basins from being overflooded and, in consequence, nuclei from being undersegmented.

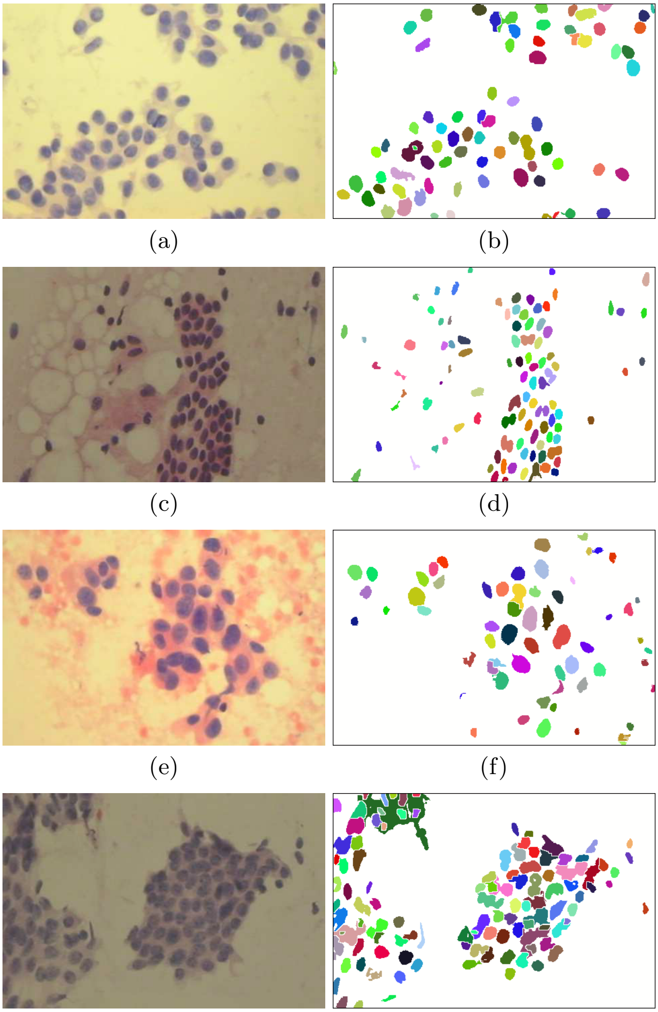
The simplified version of the watershed algorithm is given in Alg. 1. The coloring to the level of θ implements the flooding operation. It also considers a possible situation of watershed building when there is a neighbor nearby with another label. Δ defines water level increase in each iteration of the algorithm and Ψ defines the difference between the p valley's depth and the background's height in its local neighborhood.

Typical results for cytological images

Exemplary results of the presented watershed segmentation method and common errors observed in our hand-prepared benchmark database can be divided into four classes:

- *class 1*: good quality images with only small irregularities and rarely generated subbasins (a basin in another basin) (Fig. 7.9ab),
- *class 2*: errors caused by fake circles created by spots of fat (Fig. 7.9cd),
- *class 3*: mixed nucleus types: red and purple in this case and those reds which are more purple than yellow (background) are also segmented, which is erroneous (Fig. 7.9ef),
- *class 4*: poor quality image with a bunch of nuclei glued together, which causes basin overflooding and, in consequence, undersegmentation (Fig. 7.9gh).

The conducted experiments show that the watershed algorithm gives a 68.74%, on average, agreement with the hand-prepared templates using a simple XOR metric. Most errors are located at boundaries (see, for instance, Fig. 7.13a) of nuclei where the average distance between the edges of segmented and reference objects is about 3.28 pixels. The XOR metric is underestimated as a consequence of not very high level of water flooding the modeled terrain, but the shape of the nuclei seems to be preserved, which is important in our future work – the estimation of morphometric parameters of segmented nuclei.



(a) (b)

(c) (d)

(e) (f)

(g) (h)

Fig. 7.9. Exemplary results of the watershed segmentation

7.2.5 Active Contour Technique

Method description

The active contouring technique can be considered as a more advanced region growing method [44]. The algorithm groups neighboring pixels when a given homogeneity and similarity criteria is met. All joined pixels create a segment, which boundary spreads in all directions until another segment is met or the new candidates for joining introduce not acceptable error. The algorithm is stopped

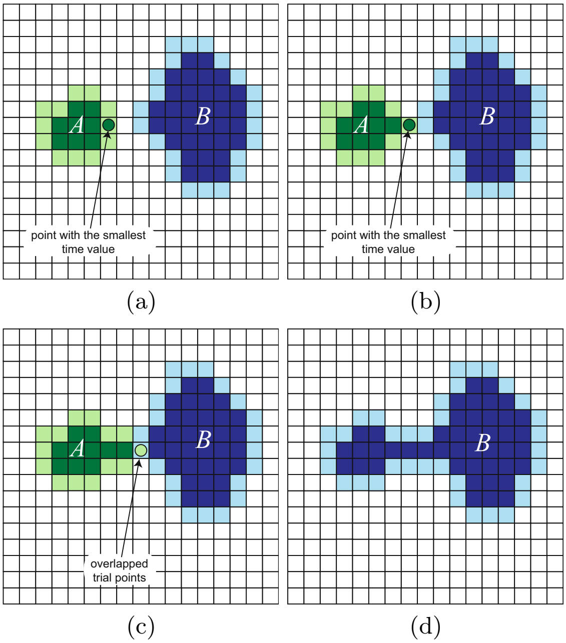


Fig. 7.10. Illustration of the contour merging operation

when all pixels get a label, that is the object in the image is separated from the background.

The images we are dealing with can contain more than a single object per image. Additionally, the assumption of the project is that the segmentation process have to be fully automatic (there is no human operator which manually initializes the method). This two factors forces us to modify the algorithm to meet the stated requirements. Thus, the algorithm, which in our case is based on fast marching method (FMM) [39], must have multilabel extension [40] and the seeding process has to be done without end-user's interaction.

In the proposed approach the multilabel FMM is initialized with a pre-segmentation mask and the results obtained from the multi-point nuclei localization stage. The background-object boundary from the pre-segmentation mask is the initial seed for the background segment. The nuclei localization points, on the other hand, are initial seeds for the object segments. The most important in this method is that the initialization mask and the nuclei localization points do not have to be perfect – all fake initial markers are fully acceptable and they do not have any influence on the final segmentation result and its quality.

The contour expansion speed of the multilabel FMM is governed globally by the function [11]:

$$F = \frac{1}{|g(x,y) - \bar{g}(i)|^3 + 1}, \tag{7.3}$$

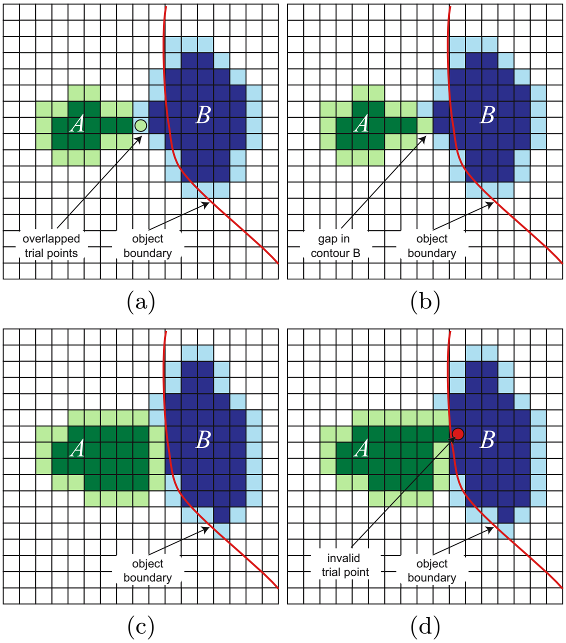


Fig. 7.11. Illustration of the contour pushing operation

where $g(x, y)$ is the color under the contour and $\bar{g}(i)$ is the mean color under the i-th segment. Such a definition of the expansion speed slows down the contour near object (nucleus) boundary. Two very close to each other spreading segments can meet while the algorithm execution. The two meeting segments can be merged (the smaller one into the bigger) when their mean color difference is below certain threshold (Fig. 7.10). The segments not classified to be merged can bush back the segment with the lower difference between considered pixel and mean color of each segment (Fig. 7.11). The pushing operation can be performed only once to reduce contour oscillation known from the classical approach and the pushed back segment can not move father at this place.

Typical results for cytological images

The conducted experiments show that the modified multilabel FMM algorithm is very stable and robust to initialization errors. Visually, segmentation quality is promising and gives good detection of even small objects (Fig. 7.12). Unfortunately, the algorithm has problems with connected nuclei and detect them as a one single object, which is erroneous. The average XOR metric score with the hand-prepared templates is only 22.32% and the average distance between the edges of segmented and reference objects is about 4.1 pixels.

Despite the mentioned problems the shape of segmented nuclei seems to be represented accurately and most errors are located at the boundaries of the

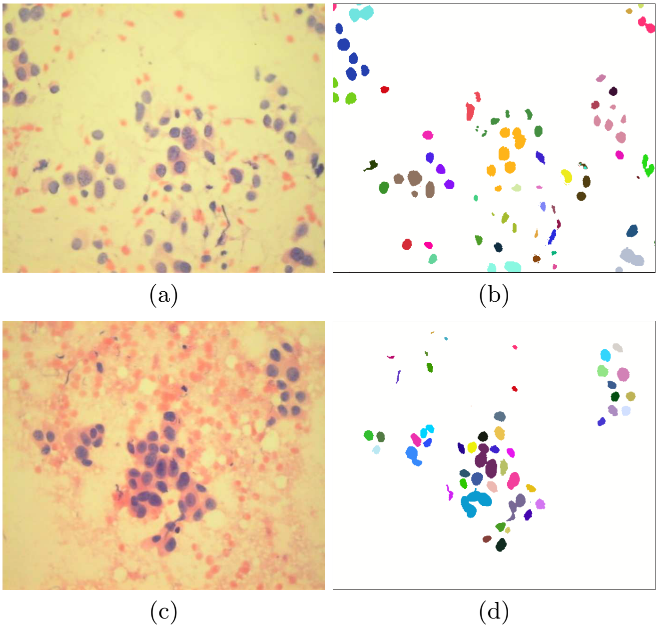

(a) (b)
(c) (d)

Fig. 7.12. Exemplary results of the active contouring segmentation

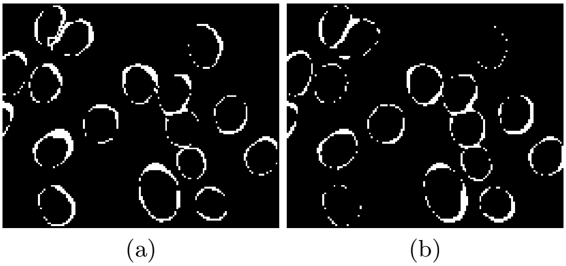

(a) (b)

Fig. 7.13. Exemplary XOR results with a fragment of the hand-prepared segmentation mask for: (a) watershed algorithm, (b) active contouring technique

segmented objects (see, for instance, Fig. 7.13b). This illustrates that the proper selection of merging threshold and detection of overlapping nuclei is still a challenge and has to be improved in the future works.

7.2.6 GrowCut Cellular Automata Segmentation

Method description

The next technique inspired by natural observations is the GrowCut cellular automata segmentation algorithm [46]. It imitates growth and struggle for domination of rivalry bacteria colonies. Each type of bacteria represents a single type of objects used in segmentation. The GrowCut algorithm was originally developed for multi-label intelligent scissors tasks for photo-editing tools. It requires manual initialization of the seed pixels, but concatenated with a proper pre-segmentation method gives a fully automated hybrid segmentation technique.

The GrowCut algorithm defines a cellular space P as $k \times m$ array, where k and m are dimensions of the image. Each of the array cells is an automaton described by the state triplet $(l_p, \theta_p, \mathbf{C}_p)$, where l_p is the label of the cell, θ_p is the *strength* of the cell and $\mathbf{C}_p$ is the feature vector of the cell defined by associated image pixel. An unlabeled image may be then considered as particular configuration state of cellular automata, where initial states for $\forall p \in P$ are set to:

$$l_p = 0, \quad \theta_p = 0, \quad \mathbf{C}_p = RGB_p, \tag{7.4}$$

where RGB_p is the three dimensional vector of pixel p color in RGB space. The final goal of the segmentation is to assign each pixel to one of the K possible labels. As stated before, we use two labels in segmentation of cytological images – the *nuclei* and the *background*.

In a single evolution step each cell (the bacteria) tries to attack all its neighbors. The evolution goal is to occupy all image area starting from a group of previously initialized pixels. Cell neighbors are defined by neighborhood system. In our approach the Moore neighborhood system was used:

$$N(p) = \Big\{ q \in Z^n : \parallel p - q \parallel_\infty := \max_{i=1,\ldots,n} |p_i - q_i| = 1 \Big\}. \tag{7.5}$$

The attack power is defined as a function of attacker q and defender p strengths and the distance between their feature vectors: $\mathbf{C}_q$ and $\mathbf{C}_p$. The basic rule of

```
// For each cell
for $\forall p \in P$
    // copy previous state
    $l_p^{t+1} = l_p^t$
    $\theta_p^{t+1} = \theta_p^t$
    // neighbors try to attack
    // current cell
    for $\forall q \in N(p)$
        if $g(\parallel \mathbf{C}_p - \mathbf{C}_q \parallel_2) \cdot \theta_q^t > \theta_p^t$
            $l_p^{t+1} = l_q^t$
            $\theta_p^{t+1} = g(\parallel \mathbf{C}_p - \mathbf{C}_q \parallel_2) \cdot \theta_q^t$
        end
    end
end
```

Algorithm 2. The GrowCut algorithm

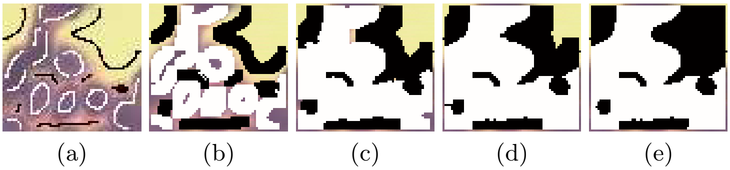


Fig. 7.14. Exemplary segmentation of a FNB image with the GrowCut algorithm. White – *nucleus* labeled cells, black – *background* labeled cells: (a) the seed, (b) step 2, (c) step 4, (d) step 6, (e) final 19-th step.

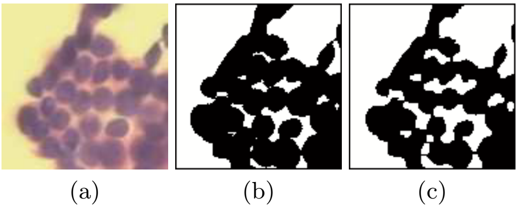


Fig. 7.15. Exemplary segmentation with the GrowCut algorithm initialized with thresholding result: (a) exemplary image, (b) thresholding result, (c) GrowCut result

automaton state change at time $t + 1$ is shown in Alg. 2. The g function is monotonous, decreasing and bounded to $[0, 1]$. For the purpose of this work, simple g function was used, as proposed in [46]:

$$g(x) = 1 - \frac{x}{\max \parallel \mathbf{C} \parallel_2}, \tag{7.6}$$

where $\max \parallel \mathbf{C} \parallel_2$ is calculated as a feature vector length for white pixel (RGB $= [255, 255, 255]$). As the strength of each cell is increasing and bounded, so the method is guaranteed to converge. Thus for any seed configuration of the image, after finite number of evolution steps, all cells are labeled and their states seize to change. Fig. 7.14 shows subsequent steps of the GrowCut segmentation for a manually initialized cytological image.

The GrowCut algorithm requires initialization of a number of cells with proper labels for each separate, consistent group of pixels (segment seed). To allow for application of the algorithm to the automated diagnostic system we employ the information from the pre-segmentation and the nuclei localization stage to initialize the seed pixels. At this point almost any rough segmentation technique (e.g. thresholding) can be also applied as the pre-segmentation (Fig. 7.15), however our research shows that initialization which leaves unclassified pixels at objects boundaries performs better. One of the techniques, which results can be utilized at the GrowCut algorithm initialization stage is the pre-segmentation mask obtained using the Hough transform. The transform result is a set of circles covering regions of the image, where nuclei are located. Pixels enclosed inside these regions are initially labeled as the *nucleus* pixels. Remaining pixels of the image are labeled as the *background*. For this type of initialization, all the image pixels are classified before the first GrowCut evolution step. The goal of the algorithm application is only to adjust the segments edges to real boundaries of objects. Therefore, to enforce the proper direction and the range of label changes within following evolution steps, associating suitable values of initial strength for both of the pixel classes is necessary.

The appropriate direction of label changes depends on the θ threshold value, used at pre-segmentation stage. For lower values of threshold, Hough transform results in a number of background pixels located in boundaries of regions labeled as *nucleus*. These pixels should change their labels do *background* in process of actual segmentation. Thus, the initial strength value of *nucleus* labeled pixels has to be less than strength of the background pixels. For higher values of the θ threshold a number of the *nucleus* pixels are incorrectly labeled as *background*. In this case labels of boundary pixels should be changed to *nucleus*. Therefore, initial strength of the *nucleus* pixels has to be greater than the *background* pixels.

The GrowCut algorithm can be also initialized with the result obtained from the multi-point nuclei localization stage described above. Due to only few initialized pixels of each segment, strengths of the cells can be set to equal values for both classes. It allows for automation of the segmentation process. However, more uninitialized pixels results in more evolution steps and so greater computational cost.

Typical results for cytological images

Typical results of the GrowCut cellular automata, initialized with the result obtained from the multi-point nuclei localization stage, can be seen in Fig. 7.16. For the exemplary image the proportion of incorrectly labeled pixels was about 6%. However, the shape of the identified nuclei segments is too ragged (due to camera sensor interlace), so additional smoothing post-segmentation stage is needed for this combination of techniques.

The problem with the Hough transform and the GrowCut cellular automata hybrid is that the optimum proportion of the initial *nucleus* pixels strength should be estimated to achieve good segmentation quality. The proportion strongly depends on the analyzed image contrast and the pigment used, so a

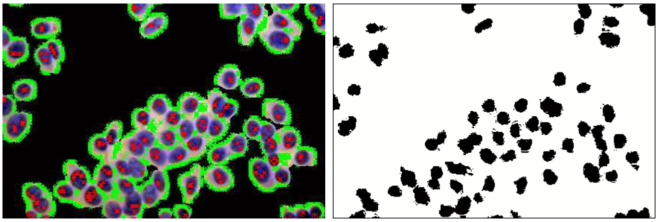

(a) (b)

Fig. 7.16. Exemplary result of the segmentation (b) with the GrowCut cellular automata, initialized with multi-point nuclei localization points (a)

potential automated diagnostic system should be learned beforehand. The second hybrid – the GrowCut cellular automata initialized with multi-point nuclei searching algorithm, can be applied with fixed initial strength values for the *nucleus* and the *background* seed pixels. However it is much more computationally expensive due to more cellular automata iterations required.

7.2.7 Fuzzy Sets of Type I and II in Thresholding

The technique used in this subsection belongs to object attribute-based methods and is based on a type-1 fuzzy thresholding technique. However type-1 fuzzy sets still have some inherent uncertainties. There are (at least) four different sources of uncertainties in type-1 fuzzy logic systems [23]: uncertainty about meanings of the words that are used (words mean different things to different people), uncertainty about consequents, uncertainty about measurements, which may be noisy and uncertainty about the data that is used to tune parameters. Thus, to address this problem type-2 fuzzy sets (T2FSs) have been formulated, which let us model and minimize the effects of this uncertainties. Such sets are fuzzy sets whose membership grades themselves are T1FSs; they are very useful in circumstances where it is difficult to determine an exact membership function for a fuzzy set; therefore, they are useful for incorporating uncertainties [15].

Nevertheless a general T2FSs computational complexity is severe and it is very difficult to justify the use of any other kind of secondary membership functions (e.g. right now there is no best choice for a T1FS, therefore secondary membership functions only complicate the matter). Thus interval T2FSs were introduced — when the T2FSs are interval T2FSs, all secondary grades equal one [23].

Another drawback of the thresholding techniques is that they are, in general, monochrome techniques. Compared to gray scale, color provides information in addition to the intensity. Color is useful or even necessary for pattern recognition and computer vision. Thus the other part of this paper is concerned with the adaptation of this monochrome technique to use extra information of color images.

Interval type-2 fuzzy sets

An interval type-2 fuzzy set (IT2 FS) $\tilde{A}$ is characterized as [24]

$$\tilde{A} = \int_{x \in X} \int_{u \in J_x \subseteq [0,1]} \frac{du dx}{ux} = \int_{x \in X} \left(\int_{u \in J_x \subseteq [0,1]} \frac{du}{u} \right) dx \Big/ x \tag{7.7}$$

where x, the *primary variable*, has domain X; $u \in U$, the *secondary variable*, has domain J_x at each $x \in X$; J_x is called the *primary membership* of x and is defined in (7.11); and, the *secondary grades* of all $\tilde{A}$ equal 1. Note that (7.7) means: $\tilde{A} : X \to \{[a,b] : 0 \leqslant a \leqslant b \leqslant 1\}$. Uncertainty about $\tilde{A}$ is conveyed by the union of all the primary memberships, which is called the *footprint of uncertainty* (FOU) of $\tilde{A}$ (see Fig. 7.17), i.e.

$$FOU(\tilde{A}) = \bigcup_{\forall x \in X} J_x = \{(x,u) : u \in J_x \subseteq [0,1]\} \tag{7.8}$$

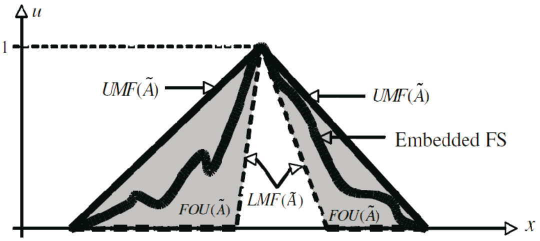


Fig. 7.17. FOU (shaded), LMF (dashed), UMF (solid) and an embedded FS (wavy line) for IT2 FS $\tilde{A}$

The upper membership function (UMF) and *lower membership function* (LMF) of $\tilde{A}$ are two type-1 membership functions that bound the FOU (Fig. 7.17). The UMF is associated with the upper bound of $FOU(\tilde{A})$ and is denoted $\overline{\mu}_{\tilde{A}}(x)$, $\forall x \in X$, and the LMF is associated with the lower bound of $FOU(\tilde{A})$ and is denoted $\underline{\mu}_{\tilde{A}}(x)$, $\forall x \in X$, i.e.

$$\overline{\mu}_{\tilde{A}}(x) \equiv \overline{FOU(\tilde{A})} \qquad \forall x \in X \tag{7.9}$$

$$\underline{\mu}_{\tilde{A}}(x) \equiv \underline{FOU(\tilde{A})} \qquad \forall x \in X \tag{7.10}$$

Note that J_x is an *interval set*, i.e.

$$J_x = \left\{(x,u) : u \in \left[\underline{\mu}_{\tilde{A}}(x), \overline{\mu}_{\tilde{A}}(x)\right]\right\} \tag{7.11}$$

so that $FOU(\tilde{A})$ in (7.8) can also be expressed as

$$FOU(\tilde{A}) = \bigcup_{\forall x \in X} \left[\underline{\mu}_{\tilde{A}}(x), \overline{\mu}_{\tilde{A}}(x)\right] \tag{7.12}$$

The upper and lower membership degrees $\overline{\mu}_{\tilde{A}}$ and $\underline{\mu}_{\tilde{A}}$ can also be defined by means of linguistic hedges like *dilation* and *concentration* on a principle membership function μ_A. Because hedges are usually available as pairs, that represent diagonally different modifications of the basic term, so it seems practical to use linguistic hedge and its reciprocal value to draw the FOU. Thus, upper and lower membership values can be defined as follows [45]:

$$\overline{\mu}_{\tilde{A}}(x) = [\mu_A(x)]^{1/\alpha}, \tag{7.13}$$

$$\underline{\mu}_{\tilde{A}}(x) = [\mu_A(x)]^{\alpha} \quad . \tag{7.14}$$

where $\alpha \in (1, \infty)$. However, according to [45] $\alpha \gg 2$ is usually not meaningful for image data.

Image thresholding with type-2 fuzzy sets

Measures of the fuzziness estimate the average vagueness in fuzzy sets. Intuitively, one should expect that if the set is maximally ambiguous then the fuzziness measure should be maximum. On the other hand, the fuzziness of the crisp set using any measure should be zero, as there is no ambiguity about whether an element belongs to the set or not. When the membership value approaches either 0 or 1, vagueness in the set decreases. Thus a fuzzy set is the most vague when $\mu_A(x) = 0.5 \; \forall x$ [34].

The most common measure of fuzziness, introduced by [17], is the *linear index of fuzziness*. For an $M \times N$ image subset $A \subseteq X$ with L gray levels $g \in [0, L-1]$, the histogram $h(g)$ and the membership function $\mu_A(g)$, the linear index of fuzziness γ_l can be defined as follows:

$$\gamma_l(A) = \frac{2}{MN} \sum_{g=0}^{L-1} h(g) \times \min[\mu_A(g), 1 - \mu_A(g)]. \tag{7.15}$$

But if images or thresholds are to be interpreted as T2FSs then there is a need for a new measurement. In this case one can ask how *ultrafuzzy* is a fuzzy set? If the degrees of the membership can be defined without any uncertainty (T1FSs), then clearly the ultrafuzziness should be minimum (=0). For the case that individual membership values can only be indicated as an interval, the amount of the ultrafuzziness should increase. And while absolutely nothing is known about the nature of membership degrees of the problem at hand, then the ultrafuzziness should be maximal (=1). With respect to these thoughts and the way a T2FS is defined, a measure of ultrafuzziness $\tilde{\gamma}$ for an $M \times N$ image subset $\tilde{A} \subseteq X$ with L gray levels $g \in [0, L-1]$, histogram $h(g)$ and membership function $\tilde{\mu}_A(g)$ can be defined as follows [45]:

$$\tilde{\gamma}(\tilde{A}) = \frac{1}{MN} \sum_{g=0}^{L-1} h(g) \times [\overline{\mu}_{\tilde{A}}(g) - \underline{\mu}_{\tilde{A}}(g)], \tag{7.16}$$

where $\overline{\mu}_{\tilde{A}}(g) = [\mu_A(g)]^{1/\alpha}$ and $\underline{\mu}_{\tilde{A}}(g) = [\mu_A(g)]^{\alpha}$, $\alpha \in (1, 2]$. This basic definition relies on the assumption that singletons sitting on the FOU are all equal in height (which is the reason why the IT2 FS is used). Thus, only the variation in the length of the FOU can be measured.

The general algorithm for the image thresholding based on type II fuzzy sets and measures of the ultrafuzziness can be formulated as follows: 1) Select the shape of the principle (skeleton) membership function $\mu_A(g)$ and initialize α; 2) Calculate the image histogram; 3) Initialize the position of the membership function; 4) Shift the membership function along the gray-level range; 5) Calculate in each position upper and lower membership values $\overline{\mu}_{\tilde{A}}(g)$ and $\underline{\mu}_{\tilde{A}}(g)$; 6) Calculate in each position the amount of the ultrafuzziness, using Eq. 7.16; 7) Locate the position g_{opt} with the maximum ultrafuzziness; 8) Threshold the image with $T = g_{opt}$.

For the thresholding algorithm to be complete, there is a need to define a suitable principle membership function. In this paper we are using following membership functions.

Function defined by [13] as follows:

$$\mu_A(g) = \begin{cases} \frac{1}{1+|g-\mu_0|/C}, \; g \leqslant T, \\ \\ \frac{1}{1+|g-\mu_1|/C}, \; g > T, \end{cases} \tag{7.17}$$

where C is a constant value such that $0.5 \leqslant \mu_A(g) \leqslant 1$, e.g. $C = g_{max} - g_{min}$; the average gray levels of the background $\mu_0 = \sum_{g=0}^{T} gh(g) \Big/ \sum_{g=0}^{T} h(g)$ and the object $\mu_1 = \sum_{g=T}^{L-1} gh(g) \Big/ \sum_{g=T}^{L-1} h(g)$, for a certain threshold T.

Color quantization

In the case of a widespread 24 bit color image representation, the number of possible colors is over 16 millon and exhaustive searching is computationally expensive or even unfeasible. However, in most cases images do not occupy the entire color gamut, and the number of colors used is much lower. Thus quantization of the color space is a viable choice.

A quantized image $M \times N$ may be regarded as a mapping defined by

$$q : M \times N \rightarrow R \subseteq \Psi \tag{7.18}$$

where $\Psi = (r, g, b)|0 \leqslant r, g, b \leqslant 255$ is the RGB color space, $R = \{\bar{r_1}, \bar{r_2}, \ldots, \bar{r_k}\}$ is a set of representative colors used in the quantized image [5].

Hence, the color quantization can be divided into two parts: color palette design, in which a desirable number of colors (usually 8–256) is specified, and pixel mapping, in which each pixel is assigned to one of the colors in the designed palette. The goal is to achieve the lowest perceivable difference between the quantized image and the original one.

The color palette design can be obtained by simply dividing a color cube into a smaller cube, but the result is usually poor. Better results are achieved by means of clustering algorithms such as K-means or fuzzy c-means. However, a major drawback of these algorithms is a high computational time. On the other hand, there are still fast quantization algorithms characterized by high quality performance and time efficiency.

When a palette has been designed, what remains is to assign the original color of each pixel in the input image to their best match in the color palette. The simplest way is to compute the distances between the original color vectors and all color vectors of the new palette, then choose the one with the minimum distance. However, faster methods can be used, such as binary tree search or *k-d* tree search.

Of course, quantization of the color space alone is not enough to be appropriate for the image segmentation. One of the reasons is that the new color palette is disordered, and therefore the histogram of the image is also chaotic and not useful for thresholding. One of the way to deal with that problem is to sort the new color palette. When sorted, the palette, and therefore the histogram, should consist of the organized data, that visually resembles the original image. In this approach the color information (palette) is discarded and the image is treated like a monochrome one, by using only the frequency of the color occurrence.

This technique requires a definition of the distance measure, such as the Euclidean distance, in order to correctly sort color vectors. However, the RGB color space is non-uniform and thus it is hard to measure color differences. Hence, other color space, like the CIE $L^*a^*b^*$ which is perceptually uniform and efficient in measuring a small color difference, can be used to sort the color palette, or even as a substitute of the RGB space for the entire color quantization process. This approach can be considered as a nonlinear projection of 3-D space onto a lower dimensional 1-D space. Also the big advantage of this method is that it can be used for any type of monochrome segmentation techniques.

Exemplary results

RGB images were converted to gray levels, for the non-fuzzy reference the Otsu technique was employed. Exemplary results obtained for techniques can be seen on Fig. 7.18. For the recursive thresholding the best results were achieved by F1 with HW MF followed closely by F2 and Otsu techniques.

Afterwards, the color quantization approach, described in Sec. 7.2.7 was utilized in order to quantize color cell images. Color images were quantized using

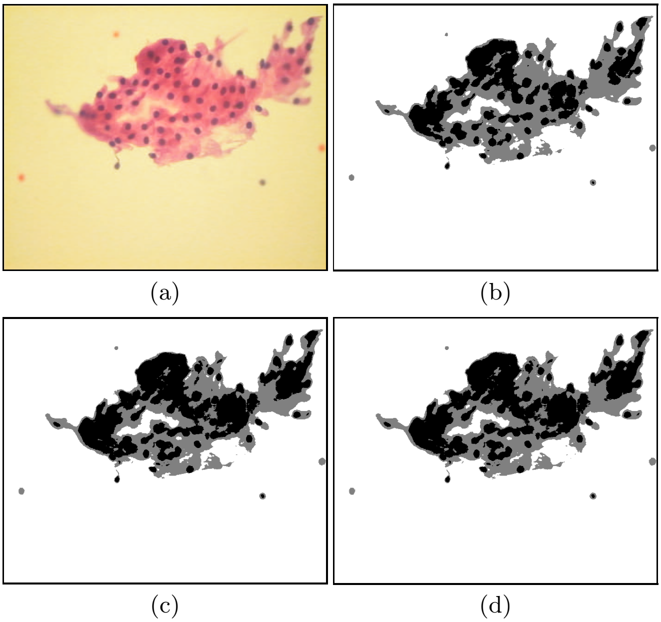

(a) (b) (c) (d)

Fig. 7.18. The comparison of different methods of conversion from RGB images to gray levels: original image (a), Otsu method (b), Fuzzy-1 method with HW MF (c) and Fuzzy-2 method with HW MF

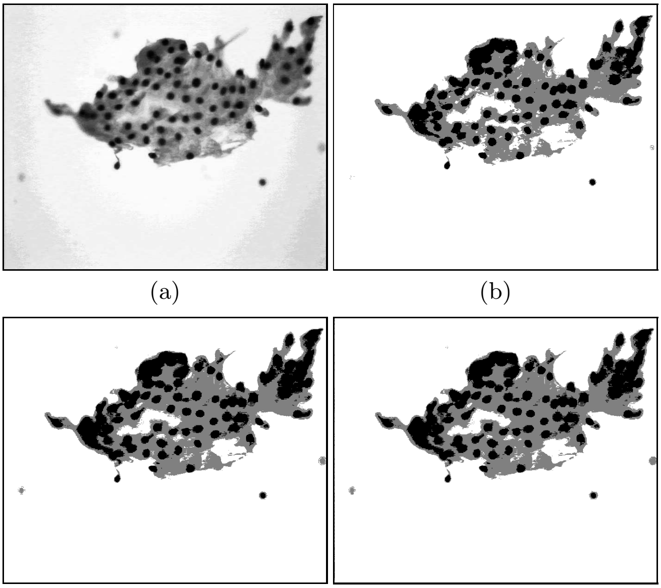


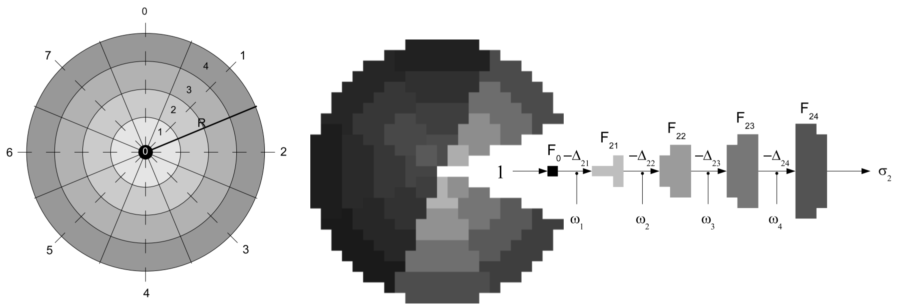
Fig. 7.19. Color quantization approach: $L^*a^*b^*$, $L^*a^*b^*$ sorted (a), Otsu method (b), Fuzzy-1 method with HW MF (c) and Fuzzy-2 method with HW MF

a minimum variance quantization method, using $L^*a^*b^*$ color spaces. Subsequently, the resulted monochrome images were thresholded using recursive thresholding techniques. The results were significantly improved over normal gray images used in previous example. Otsu technique and fuzzy based techniques, with the HW membership function, achieved comparable score with minor differences. Exemplary results obtained for this techniques can be seen on Fig. 7.19.

The exact results can be seen in [7].

7.2.8 The Sonar-Like Segmentation Method

The sonar-like segmentation is a novel method developed by the authors for the cytological image segmentation purpose. The method consists in classification of image pixels based on spatial analysis of a pixel feature variance. Each class represents a visual artefact (e.g. edges, uniform inner regions, etc.). Artefact classes are distinguished by comparison with a number of feature variance templates. The result of pixel classification is a set of regions which after proper merging allows for identification of actual objects. The name of the method originates from a similarity of the feature variance analysis mechanism to the physical phenomenon of a sonar sound wave speed alternation between water regions of different physical condition (temperature, pressure, etc.).



(a) (b)

Fig. 7.20. Sonar neighborhood model for $s = 8$, $r = 4$ and $\theta = 0$ (a) and its discrete realization for range $R = 8$ (b)

The base concept for the feature variance analysis is the sonar pixel neighborhood. It is defined as a set of concentrically situated rings of pixels. All of the rings have the same width and together they cover a circle, which centre is in the examined pixel. The circle is then partitioned into a number of slices along concentrically located bearings. Intersections of the rings and the slices are the neighborhood sectors. The number of sectors in each ring is the same, and it equals to the number of the neighborhood bearings.

Formally, the sonar neighborhood N_{sonar} can be defined as quadruplet:

$$N_{sonar} = (R, r, s, \theta) \tag{7.19}$$

where R is the neighborhood range, r is the number of rings, s is the number of bearings (sectors in each ring) and θ is the angle between the zero-sector axis and the 'north' bearing. The neighborhood bearings are numbered from 0 and rings are numbered from 1. The zero-ring is the examined pixel itself. Figure 7.20 shows a visual model of the sonar neighborhood for $s = 8$, $r = 4$ and $\theta = 0$ (a) and its discrete realization for $R = 8$ (b).

The sonar neighborhood applied to an actual pixel allows for calculation of the sonar vector $\boldsymbol{\sigma}$. The length of the vector equals to the number of bearings, so each of the vector values represents the variance of the examined feature along single bearing. The values bounded to [0, 1] are calculated on the basis of a set of differences of the feature statistics between subsequent sectors along the bearing. The statistics used in the presented application of the method was arithmetic mean of the luma component of the YCbCr color model.

Figure 7.20(b) shows a scheme of a sonar vector value calculation for a single bearing. The value can be seen as the energy of the sound wave signal after passing from the central point through subsequent sectors along bearing. Passage through each boundary between two adjacent sectors decreases the signal energy by Δ_{si} value, which is calculated as a weighted function of the sectors feature statistics difference:

$$\Delta_{si} = \delta(|f_{s,i} - f_{s,i-1}|)w_i \tag{7.20}$$

where $f_{s,i}$ is the feature statistics of the sector at s-bearing of the i-th ring, and w_i is the weight for the boundary between $i-1$-th and i-th ring. The function δ is a monotonous nondecreasing function, bounded to [0, 1]. The function can be defined as follows:

$$\delta(t) = \begin{cases} \frac{t}{d} \; if \; t \leq d, \\ 1 \; if \; t > d, \end{cases} \tag{7.21}$$

where $d > 0$ is the sensitivity parameter defining the threshold value of the statistics difference over which the value of δ function remains 1. If the feature variance in the neighborhood is high enough, the value can reach zero just after subtracting only few of the delta values. If so, subsequent feature differences are ignored and the value remains zero.

The weights for subsequent boundaries are inversely proportional to the distance of the boundary to the central point, so differences between sectors closest to the neighborhood centre have the greatest impact on the sonar vector value. The weights sequence can be arbitrarily chosen or defined as a function of the ring number, for example:

$$w_i = \frac{1}{2^i w} \tag{7.22}$$

where w is the arbitrary weighting factor.

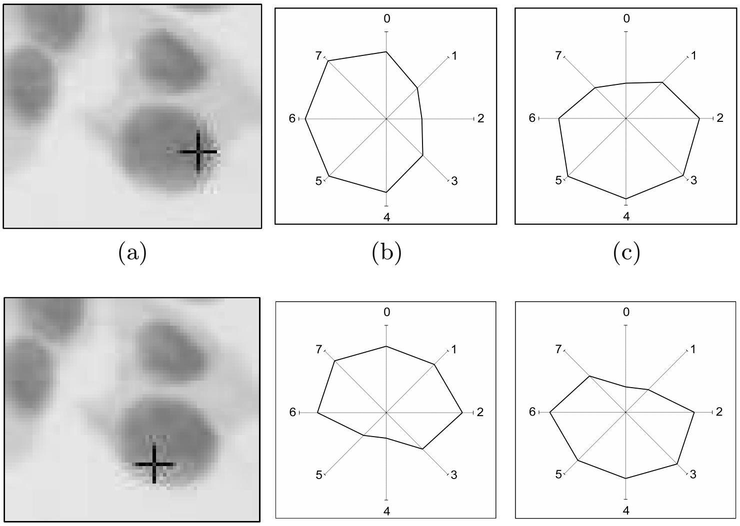
The calculated sonar vector can be visualized with a radar plot. A shape of the plot expresses the feature variance in the neighborhood of the examined pixel. Later in this section, the sonar vectors visualizations are stated as sonar views.

The sonar vector matrix representing the image, prepared as described above, is used for a classification of the image pixels. The classification is performed by the comparison of the pixel sonar vector with a number of template vectors. The template vectors can be prepared manually or calculated automatically by the evaluation of average sonar vectors for a set of images classified with reference segmentation masks.

Some of the searched artifacts, like edges, have a number of same-shaped, but rotated sonar vectors. To reduce computational cost regarding introduction of a number of sub-classes for single artefact, a normalization of sonar vectors can be performed beforehand. The normalization process is a cyclic rotation of the vector values, until the lowest value is at the zero-bearing. After such an operation, each of the artefact sub-class sonar vectors equals the vector of the unrotated artefact with lowest value at the zero-bearing. So, a single class template can be produced. Figure 7.21 shows sonar views and normalized sonar views for pixels located at differently aligned nucleus edges.

Actual classification of the image pixels is preformed by finding the minimal mean squared error between the pixel sonar vector and each of the template vectors. The mean squared error MSE of the sonar vector compared to the C-class template vector is defined as follows:

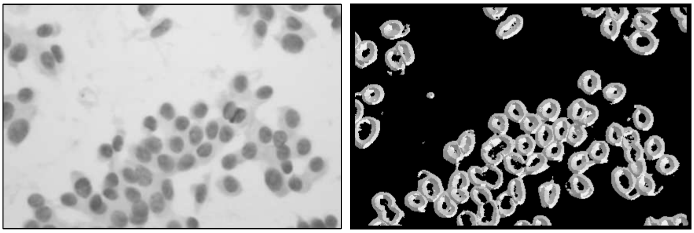
$$MSE_C(\boldsymbol{\sigma}) = E((\boldsymbol{\sigma} - \boldsymbol{\sigma}_{\mathbf{C}})^2) = \frac{1}{s} \sum_{i=0}^{s} (\sigma_i - \sigma_{Ci})^2 \tag{7.23}$$



(a) (b) (c)

(d) (e) (f)

Fig. 7.21. Sonar views and normalized sonar views for pixels located at differently aligned nucleus edges: (a), (d) - location of pixels; (b), (e) - sonar views; (c), (f) - normalized sonar views



(a) (b)

Fig. 7.22. Cytological image (a) and the result of the pixel classification with Sonar (b)

where $\boldsymbol{\sigma}$ is the sonar vector, $\boldsymbol{\sigma}_C$ is the sonar vector template of the class C, and s is the number of bearings. The pixel is labeled with the label of the class with the minimal mean squared error. Figure 7.22 shows a sample cytological image and the result of the pixel classification with Sonar. For the 'edge' class the saved number of the rotation steps is marked with the greyscale, from white color for the bearing 0 to dark grey for the bearing 7.

At this stage all artifacts are identified and located. For cytological FNB images the expected result is an identification and location of nuclei. Each of the nuclei consists of a single interior and a number of adjacent edge-artifacts. To achieve the proper segmentation, an additional step of artifacts merging needs to be performed.

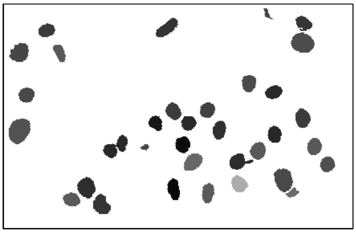


Fig. 7.23. Result segmentation of the cytological image

As can be observed in the Figure 7.22, actual edges of objects are surrounded from both sides with the edge segments. One side is the nucleus boundary, and the other is the background boundary. Adjacent segments of the actual edge have opposite rotation, so it can be distinguished which neighboring interior segment should be merged with the edge segment. Along a single boundary, the edge segments should be merged if the distance between their rotations equals 1. The segments should be also merged with the interior segment located below the 'north' edge. The result of the segment merging process is the segmentation of the image along actual edges of objects (Fig. 7.23).

7.2.9 Decision Tree Method

Another algorithm of the pixel-based segmentation area is the decision tree-based method of pixel classification. The primary mechanism of the method is classification of image pixels with a decision tree, which input is the pixel color, and the output is the probability of the pixel membership in each of the applied object-classes. It is assumed that the numerical information on a pixel color (e.g. RGB color components) is sufficient to identify the pixel as a member of one of a number of object-classes.

For the purpose of nuclei segmentation in cytological images, three classes are introduced: the nucleus (N), the background (B) and the inter-nucleus (I) class. The last of the three classes represents all objects which cannot be unambiguously labeled as a member of one of the former classes. These pixels are elements of erythrocytes and cytoplasm.

The decision tree training process is based on a set of manually prepared three-color image masks along with regarding training images. All image masks classify a set of pixels to the three classes. Classified pixels of all the training images together are used as the information for the decision tree training process. The result of the process is a decision tree, which assigns probabilities of the pixel being a member of each class. Due to no spatial information is used in the training process there is no requirement for the masks of images to be completed. Some of the mask pixel can be left unlabeled. It allows excluding ambiguous pixels from the decision tree training process. Figure 7.24 shows a sample training image (a) with regarding mask (b).

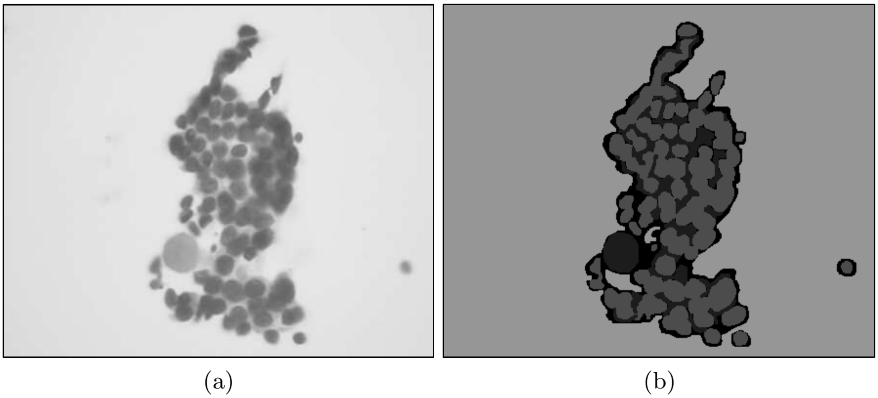

(a) (b)

Fig. 7.24. Sample training image (a) with regarding mask (b). Red - the nucleus pixels; green - the background pixels; blue - inter-nucleus pixels; black - unclassified (ambiguous) pixels

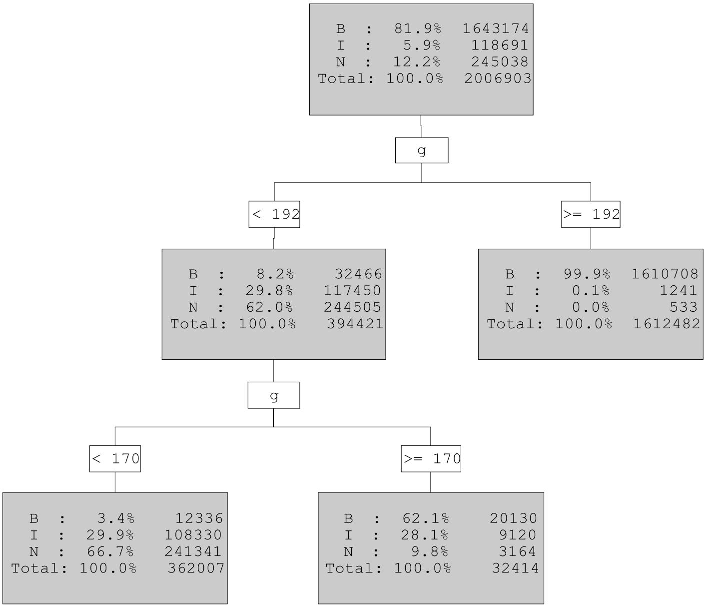


Fig. 7.25. Sample decision tree with 3 leaves

As stated before, the output of the decision tree for a pixel is a set of probabilities of the pixel membership in each class. Figure 7.25 presents a sample decision tree with 3 leaves.

The output of the decision tree can be visualized by rendering the image with pixel color components proportional to the respective probabilities. Using the same components for classes as in mask images, one can get output images similar to the masks, however there will be no unlabeled pixels left. Actual values of color components for pixels can be calculated as follows:

$$C = P(O)C_{max} \tag{7.24}$$

where C is the color component, $C \in \{R, G, B\}$, $P(O)$ is the probability of the class O membership, $O \in \{N, I, B\}$, C_{max} is the maximum value of a single color component (255 for 24-bit color RGB images). Figure 7.26 shows a sample output image.

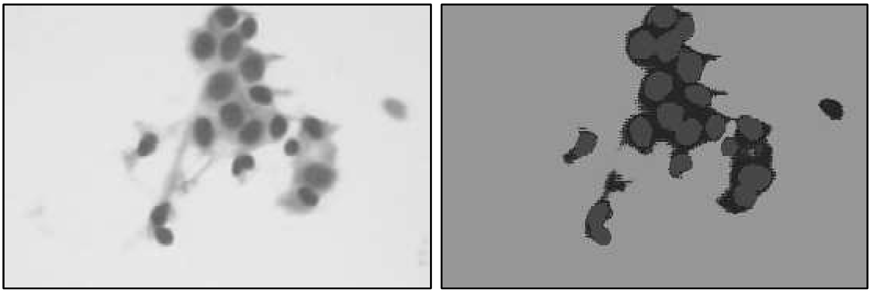

(a) (b)

Fig. 7.26. Sample result of the decision tree application: input image (a), output image (b)

The number of combinations of class-membership probabilities in the decision tree output is much lower than the number of RGB color components in the processed image. Output images have the maximum number of distinct colors equal to the number of probability combinations. So the application of the decision tree can be perceived as the problem complexity reduction technique for another segmentation algorithm. Due to the output of the method are rendered images along with probability matrices, almost any segmentation technique can be applied on the actual segmentation stage. For example the thresholding with 3D homograms can be applied for the actual segmentation. In the authors' research the decision tree prepared with the SAS Enterprize Miner had 21 to 56 leaves. The number of leaves equals the number of bins of a homogram needed to perform the segmentation, so the decision tree can play the role of color quantizer based on the information on the image color characteristics.

7.3 Conclusions

In this chapter we bring together the latest results from researchers involved in state-of-the-art work in cytological image segmentation, providing both a survey on segmentation well-known techniques supporting such processes as measurement, visualization, registration and reconstruction of image and a collection of new approaches elaborated for the case of cytopathologic scans. A wide variety of new methods is presented, including solutions based on fuzzy sets of types I and II, clustering, decision trees, shape detection, active contours and many others as well as they hybrids. Issues of automated segmentation of cell nuclei are broadly described on the examples of microscopic cytological images obtained via fine needle biopsy technique. Although some of the predictions would probably be shared by many people working in this field, this presentation still will be subjective and personal. In our opinion, perspectives for further development of cytological image segmentation are closely connected with computational intelligence, closer interaction between system and a human operator as well as semantic image interpretation.

References

1. Ballard, D.: Generalizing the Hough transform to Detect Arbitrary Shapes. Pattern Recogn 13(2), 111–122 (1981)
2. Boldrini, J., Costa, M.: An Application of Optimal Control Theory to the Design of Theoretical Schedules of Anticancer Drugs. Int. J. Appl. Math. and Comput. Sci. 9(2), 387–399 (1999)
3. Carlotto, M.: Histogram analysis using a scale space approach. IEEE Trans. Pattern Analysis and Machine Intelligence 9(1), 121–129 (1987)
4. Chen, C., Luo, J., Parker, K.: Image segmentation via adaptive K-mean clustering and knowledge-based morphological operations with biomedical applications. IEEE Trans. Image Processing 7(12), 1673–1683 (1998)
5. Cheng, H.D., Chen, C.H., Chiu, H.H., Xu, H.: Fuzzy homogeneity approach to multilevel thresholding. IEEE Trans. Image Processing 7(7), 1084–1088 (1998)
6. Duda, R., Hart, P.: Use of the Hough Transformation to Detect Lines and Curves in Picture. Comm. ACM 15, 11–15 (1972)
7. Dziekan, Ł., Marciniak, A., Obuchowicz, A.: Segmentation of color cytological images usin type II fuzzy sets. In: Korbicz, J., Patan, K., Kowal, M. (eds.) Fault Diagnosis and Fault Tolerant Control, pp. 263–270. Academic Publishing House EXIT, Warszawa (2007)
8. Fu, K.S., Mui, J.K.: A survey on image segmentation. Pattern Recogn 13, 3–16 (1981)
9. Hrebień, M., Nieczkowski, T., Korbicz, J., Obuchowicz, A.: The Hough transform and the GrowCut method in segmentation of cytological images. In: Proc. Int. Conf. Signals and Electronic Systems ICSES 2006, Łódź, Poland, vol. 1, pp. 367–370 (2006)
10. Hrebień, M., Korbicz, J., Obuchowicz, A.: Hough transform (1+1) search strategy and watershed algorithm in segmentation of cytological images. In: Proc. 5th Int. Conf. Comp. Recogn. Systems CORES 2007. Adv. in Soft Computing, vol. 45, pp. 550–557. Springer, Heidelberg (2007)
11. Hrebień, M., Steć, P.: Fine Needle Biopsy Material Segmentation with Hough Transform and Active Contouring Technique. Journal of Medical Inform. & Techn. 10, 25–34 (2007)
12. Gonzalez, R., Woods, R.: Digital Image Processing. Prentice-Hall, Englewood Cliffs (2002)
13. Huang, L.K., Wang, M.J.: Image thresholding by minimizing the measure of fuzziness. Pattern Recogn. 28, 41–51 (1995)

14. Jain, A.K.: Fundamentals of Digital Image Processing. Prentice-Hall, Englewood Cliffs (1989)
15. Karnik, N.N., Mendel, J.M., Liang, Q.: Type-2 fuzzy logic systems. IEEE Trans. Fuzzy Syst. 7, 643–658 (1999)
16. Kass, M., Witkin, A., Terauzopoulos, D.: Snakes: active contour models. In: Proc. 1st Int. Conf. on Computer Vision, pp. 259–263 (1987)
17. Kaufmann, A.: Introduction to the Theory of Fuzzy Subsets—Fundamental Theoretical Elements. Academic Press, New York (1975)
18. Kimmel, M., Lachowicz, M., Świerniak, A. (eds.): Cancer Growth and Progression, Mathematical Problems and Computer Simulations. Int. J. of Appl. Math. and Comput. Sci. 13(3) (2003) (special Issue)
19. Lee, M., Street, W.: Dynamic learning of shapes for automatic object recognition. In: Proc. 17th Workshop Machine Learning of Spatial Knowledge, pp. 44–49 (2000)
20. Madisetti, V., Williams, D.: The Digital Signal Processing Handbook. CRC Press, Boca Raton (1997)
21. Marciniak, A., Monczak, R., Kołodziński, M., Prętki, O.A.: Test base for the breast cancer diagnosis using FNB method. In: Proc. Nat. Conf. Artificial Intelligence in Biomedical Engineering SIIB 2004, Kraków, Poland, [4] CD-ROM (2004) (in Polish)
22. Marciniak, A., Obuchowicz, A., Monczak, R., Kołodziński, M.: Cytomorphometry of Fine Needle Biopsy Material from the Breast Cancer. In: Proc. 4th Int. Conf. Comp. Recogn. Systems CORES 2005. Adv. in Soft Computing, pp. 603–609. Springer, Heidelberg (2005)
23. Mendel, J.M., John, R.I.: Type-2 Fuzzy Sets Made Simple. IEEE Trans. Fuzzy Systems 10(2), 117–127 (2002)
24. Mendel, J.M.: An architecture for making judgments using computing with words. Int. J. Appl. Math. Comput. Sci. 12(3), 325–335 (2002)
25. Michalewicz, Z.: Genetic Algorithms + Data Structures = Evolution Programs. Springer, Heidelberg (1996)
26. Nevatia, R.: Image Segmentation. In: Young, T.Y., Fu, K.S. (eds.) Handbook of Pattern Recognition and Image Processing. Academic Press, NY (1986)
27. Nieczkowski, T., Obuchowicz, A.: Application of cellular automaton for enhancing segmentation results of breast cancer fine needle biopsy microscope images. In: Kłopotek, M., Tchórzewski, J. (eds.) Proceedings of Artificial Intelligence Studies, vol. 3, pp. 71–78. University of Podlasie Press, Siedlce (2006)
28. Nieczkowski, T., Obuchowicz, A.: 'Sonar' - Region of Interest Identification and Segmentation Method for Cytological Breast Cancer Images. In: Proc. 5th Int. Conf. Comp. Recogn. Systems CORES 2007. Adv. in Soft Computing, vol. 45, pp. 566–573. Springer, Heidelberg (2007)
29. Nieczkowski, T., Obuchowicz, A.: Application of decision trees to filtering and segmentation of breast cancer fine needle biopsy microscope images. Biocybernetics and Biomedical Engineering 27(4), 59–70 (2007)
30. Obuchowicz, A.: Evolutionary Algorithms for Global Optimization and Dynamic System Diagnosis. Lubuskie Scientific Society Press, Zielona Góra (2003)
31. Obuchowicz, A., Korbicz, J.: Evolutionary methods in designing diagnostic systems. In: Korbicz, J., Kościelny, J.M., Kowalczuk, Z., Cholewa, W. (eds.) Fault Diagnosis: Models, Artificial Intelligence, Applications, pp. 301–331. Springer, Heidelberg (2004)
32. Olabarriaga, S.D., Smeulders, A.W.M.: Interaction in the segmentation of medical images: A survey. Medical Image Analysis 5, 127–142 (2001)

33. Otsu, N.: A threshold selection method from grey-level histograms. IEEE Trans. Systems, Man and Cybernetics 9(1), 62–66 (1979)
34. Pal, N.R., Bezdek, J.C.: Measures of fuzziness: a review and several new classes. In: Yager, R.R., Zadeh, L.A. (eds.) Fuzzy Sets, Neural Networks and Soft Computing, Van Nostrand Reinhold, New York (1994)
35. Pena-Reyes, C., Sipper, M.: Envolving fuzzy rules for breast cancer diagnosis. In: Proc. Int. Symp. on Nonlinear Theory and Application, vol. 2, pp. 369–372. Polytechniques et Universitaires Romandes Press (1998)
36. Pratt, W.K.: Digital image processing. Wiley, Chichester (2001)
37. Russ, J.: The Image Processing Handbook. CRC Press, Boca Raton (1999)
38. Setiono, R.: Extracting rules from pruned neural networks for breast cancer diagnosis. Artificial Intelligence in Medicine, 37–51 (1996)
39. Sethian, J.: Fast marching methods. SIAM Review 41(2) (1999)
40. Steć, P., Domański, M.: Video Frame Segmentation Using Competitive Contours. In: Proc. 13th European Signal Processing Conference, Antalya, Turkey (2005)
41. Street, W.: Xcyt: A system for remote cytological diagnosis and prognosis of breast cancer. In: Jain, L. (ed.) Soft Computing Techniques in Breast Cancer Prognosis and Diagnosis, pp. 297–322. World Scientific Publishing, Singapore (2000)
42. Su, M., Chou, C.: A modified version of the K-means algorithm with a distance based on cluster symmetry. IEEE Trans. Pattern Analysis and Machine Intelligence 23(6), 674–680 (2001)
43. Świerniak, A., Ledzewicz, U., Schättler, H.: Optimal Control for a Class of Compartmental Models in Cancer Chemotherapy. Int. J. of Appl. Math. and Comput. Sci. 13(3), 357–368 (2003)
44. Tadeusiewicz, R.: Vision Systems of Industrial Robots. WNT, Warszawa (in Polish) (1992)
45. Tizhoosh, H.R.: Image thresholding using type II fuzzy sets. Pattern Recognition 38, 2363–2372 (2005)
46. Vezhnevets, V., Konouchine, V.: “GrowCut” – Interactive Multi-Label N-D Image Segmentation by Cellular Automata. In: Proc. 15th Int. Conf. on Comp. Graphics and Appl. GraphiCon 2005, Novosibirsk, Russia, pp. 150–156 (2005)
47. Vincent, L., Soille, P.: Watersheds in digital spaces: an efficient algorithm based on immersion simulations. IEEE Trans. Pattern Analysis and Machine Intelligence 13(6), 583–598 (1991)
48. Wolberg, W., Street, W., Mangasarian, O.: Breast cytology diagnosis via digital image analysis. Analytical and Quantitative Cytology and Histology 15(6), 396–404 (1993)
49. Zhou, P., Pycock, D.: Robust statistical models for cell image interpretation. Image and Vision Computing 15(4), 307–316 (1997)
50. Zouagui, T., Benoit-Cattin, H., Odet, C.: Image segmentation functional model. Pattern Recognition 37(9), 1785–1795 (2002)
51. Żorski, W.: Image Segmentation Methods Based on the Hough Transform. Studio GiZ, Warszawa (in Polish) (2000)