Semi-Supervised Learning of Edge Filters for Volumetric Image Segmentation

Margret Keuper[1,2], Robert Bensch[1,2], Karsten Voigt[2,3], Alexander Dovzhenko[2,3], Klaus Palme[2,3,4], Hans Burkhardt[1,2], Olaf Ronneberger[1,2]

[1]Lehrstuhl für Mustererkennung und Bildverarbeitung,
Institut für Informatik, [2]Centre of Biological Signalling Studies (BIOSS),
[3]Institut für Biologie II, [4]Freiburg Inst. for Advanced Studies (FRIAS),
Albert-Ludwigs-Universität Freiburg
keuper@informatik.uni-freiburg.de

Abstract. For every segmentation task, prior knowledge about the object that shall be segmented has to be incorporated. This is typically performed either automatically by using labeled data to train the used algorithm, or by manual adaptation of the algorithm to the specific application. For the segmentation of 3D data, the generation of training sets is very tedious and time consuming, since in most cases, an expert has to mark the object boundaries in all slices of the 3D volume. To avoid this, we developed a new framework that combines unsupervised and supervised learning. First, the possible edge appearances are grouped, such that, in the second step, the expert only has to choose between relevant and non-relevant clusters. This way, even objects with very different edge appearances in different regions of the boundary can be segmented, while the user interaction is limited to a very simple operation. In the presented work, the chosen edge clusters are used to generate a filter for all relevant edges. The filter response is used to generate an edge map based on which an active surface segmentation is performed. The evaluation on the segmentation of plant cells recorded with 3D confocal microscopy yields convincing results.

1 Introduction

The segmentation of volumetric data is a difficult, some say ill-posed problem. Depending on the specific application as well as on the imaging technique, the desired object boundary can have very different appearances. Thus, for every new problem setting, the used method needs to be adapted and special prior knowledge about the application has to be included. Important questions can be: What is the appearance of the objects boundary? What edges are we looking for? What is the texture of the object? In the same dataset, e.g. of a plant cell, the user could be looking for either outer (the cell wall) or inner borders (plasmalemma), or intracellular compartments (as the nucleus or chloroplasts). This information can either be acquired by learning from ground truth data (e.g. in [1]), if sufficient labeled data is available, or learned from user interaction. In

most cases of 3D image analysis, the generation of ground truth segmentations (needed as input for the learning step) is a very tedious work, because the expert has to draw correct object boundaries in every single slice of the volume. This is why we are presenting a semi-supervised, user-guided segmentation method. In [2], a user-guided tool for the segmentation of medical data is presented. There, the authors propose a twofold strategy: they create a graph description of contour fragments with a tesselation of the image plane. The actual segmentation is formulated as a path optimization, where the user has to manually select control points on the contour. In [3], a user guided level set segmentation is presented, that allows the user, similarly to our method, to define the edge map before starting a level set segmentation. The edge map is defined by a threshold either on the data itself or on the gradients. Thus, the method works satisfyingly only if the edge information is homogeneous over the whole dataset.

We are presenting a segmentation framework, that uses K-means clustering of the original object edges in order to enable the user to choose between different possible edge appearances in one sample dataset. This information is used to design an edge filter for the entire database, that can handle different appearances of an object's boundary. Then, a first segmentation with active surfaces is performed, using the force field derived from the edge filter response. The filter can be refined by adding more training samples. The needed user interaction is a very simple, quick and intuitive operation.

Since we are working on biological cell data, we have adapted the presented framework to the segmentation of star-shaped objects. The evaluation was performed on 3D confocal recordings of developing plant cells.

2   Framework

The general workflow of the presented method is displayed in figure 1. We assume that in the given database of recorded objects, the object detection step is already solved and for every object, the estimated position of the center $\mathbf{c}$ is given. For spherical objects, this detection step can be performed using the Hough transform as e.g. in [4]. Given this set of objects, the first step is to choose a random sample dataset. In this dataset, we find candidate positions at which edge profiles are extracted. These edge profiles are used as features and are grouped into different clusters. The processing up to this step will be presented in detail in 2.1. The result of the clustering is mapped into the original dataset: edges belonging to the same cluster are displayed in the same color. The next and most important step is the user interaction. From the displayed edge distributions presented in a 3D slice viewer, the user can decide which edges lie on the desired object contour. This information is used to design a filter for the specified edges, which is applied to the entire database. The generation of the filter is described in 2.2. The filter response is used to perform a parametric 3D active surface segmentation using spherical harmonics. This step will be described in 2.3. The user can now verify the resulting segmentations and, if it is not sufficient for

all datasets, choose the next training sample. The edge appearances in this new sample are used to refine the filter and accordingly, to refine the segmentation.
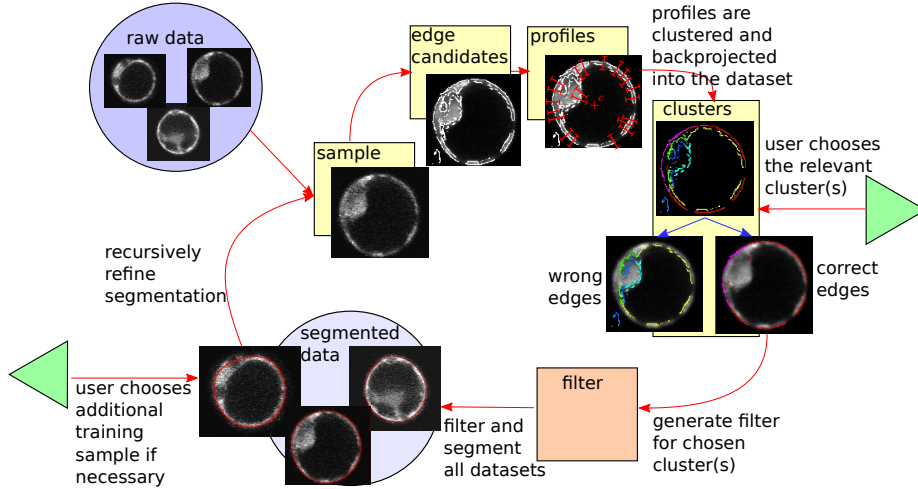

Fig. 1. Schematic overview over the whole workflow.

2.1 Profile Extraction and Grouping

In most applications, the user is interested in laying a boundary on certain positions with high gradient magnitude, i.e. image edges. In positions where the information is lacking, the user usually wants a smooth interpolation of the boundary. As candidate positions at which the profiles will be extracted, we thus choose points with high gradient magnitude. To avoid finding too many candidates and to ensure that we are looking at the most important positions in the sample, we perform a non maximum suppression in gradient direction and take all the maxima in gradient magnitude as candidates. At these positions grayvalue profiles are extracted in radial direction from the center. This makes sense because we are assuming that the objects are star-shaped (i.e. there exists a point c such that each line segment connecting c to the object's boundary lies completely within the shape). When looking at more general shapes with spherical topology, one should extract profiles that are normal to the surface at this position, instead. For the extraction of the profiles, two parameters can be adjusted: the step size of the profile and its length. These have to be chosen such that the desired edge appearance can be captured and resolved.

We want to use these profiles as features to describe the appearance at the respective position. For microscopic data we are expecting strong variations in the absolute grayvalues even within the same recording due to absorption. To make our features robust against these variations, we use the derivative of the

profile grayvalues, which is invariant against a graylevel offset. The continuous profile derivative p is a function of the radial length $l \in \mathbb{R}$ and the position $\mathbf{x}$. With respect to the center $\mathbf{c}$, $p(\mathbf{x}, l)$ is given by

$$p(\mathbf{x}, l) = \frac{\mathrm{d}}{\mathrm{d}l} I \left(\mathbf{x} + l \cdot \frac{\mathbf{x} - \mathbf{c}}{|\mathbf{x} - \mathbf{c}|} \right), \tag{1}$$

where I is the sample dataset. Thus, the discrete profile vector $\mathbf{p_x}$ is given by $\mathbf{p_x}(i) = p(\mathbf{x}, \lambda \cdot (i - \frac{L}{2}))$, where $0 \leq i < L \in \mathbb{N}$ is the position on the profile, L is the profile length and λ is the stepsize. In order to be also invariant against multiplicative grayvalue changes, we normalize the features by the maximum absolute value.

$$\bar{\mathbf{p}}_{\mathbf{x}}(i) = \frac{\mathbf{p_x}(i)}{\max_i(|\mathbf{p_x}(i)|)} \tag{2}$$

The $\bar{\mathbf{p}}_{\mathbf{x}}$ are clustered using K-means clustering (see e.g. [5]). This basic clustering method is suitable for our purpose, because it directly measures the similarity between the profiles and discriminates as soon as the euclidean distance is too large. The number of clusters scales with the number of extracted profiles. In our implementation, there are on average 4000 profiles in one cluster.

2.2   User Specified Filtering

Once the profiles are grouped into different clusters C_j, the user has to decide which of the clusters are relevant for the actual application. Therefore, the data must be presented such that it is easy to distinguish between relevant and non relevant clusters. For every cluster, we generate a 3D overlay plot in which all the points belonging to this cluster are marked in the original 3D sample data. The user can view this overlay and decide, whether the marked points lie on the desired boundary or not. This information is used to generate a Gaussian probability density function (PDF) for every chosen cluster, with

$$f_{C_j}(\bar{\mathbf{p}}) = \frac{1}{(2\pi)^{l/2}|\Sigma_{C_j}|^{1/2}} \cdot \mathrm{e}^{-\frac{1}{2}(\bar{\mathbf{p}} - \boldsymbol{\mu}_{C_j})^T \Sigma_{C_j}^{-1}(\bar{\mathbf{p}} - \boldsymbol{\mu}_{C_j})}, \tag{3}$$

where Σ_{C_j} is the covariance matrix of all profiles $\bar{\mathbf{p}}_{C_j}$ in the chosen cluster C_j and $\boldsymbol{\mu}_{C_j}$ the expected value. With these PDFs, the edge filter is already defined and can be applied to all objects in the database. In the filtering step, we have to extract the profile $\bar{\mathbf{p}}_{\mathbf{x}}$ at every position $\mathbf{x}$ in the dataset. For this profile, we compute the Mahalanobis distance

$$D_{\mathrm{M}}(\bar{\mathbf{p}}_{\mathbf{x}}, C_j) = \sqrt{\left(\bar{\mathbf{p}}_{\mathbf{x}} - \boldsymbol{\mu}_{C_j} \right)^T \Sigma^{-1} \left(\bar{\mathbf{p}}_{\mathbf{x}} - \boldsymbol{\mu}_{C_j} \right)} \tag{4}$$

to every chosen cluster. The filter response is then given by

$$A(\mathbf{x}) = \min_{C_j} \left(D_{\mathrm{M}}(\bar{\mathbf{p}}_{\mathbf{x}}, C_j) \right). \tag{5}$$

A has low values, where the distance to the closest cluster center is small, i.e. where the profiles are similar to those belonging to the selected clusters. In order to have high responses at these positions, we compute $\bar{A} = 1 - A/\max(A)$. $\bar{A}$ is used to generate the external force field for the parametric active surfaces.

2.3 Parametric Active Surfaces

Active surfaces are a common tool for the segmentation of 3D biological data. Given a rough estimate of the objects position and size, i.e. its center $\mathbf{c}$ and radius r, an accurate fitting of the model to the underlying data can be performed. Active surfaces classically have internal energies E_{int}, depending only on the shape of the model itself, and are exposed to external energies E_{ext} coming from the underlying dataset. The total energy of an active surface s is thus $E(s) = E_{\text{int}}(s) + E_{\text{ext}}(s)$. The active surface adaptation, i.e. the minimization of $E(s)$, leads to an Euler-Lagrange equation that can be considered as a force balance system $\mathbf{F}_{\text{int}} + \mathbf{F}_{\text{ext}} = \mathbf{0}$ (see [6]). It can be performed in the spatial domain, using a suitable surface mesh as it was done e.g. in [7] or in the Spherical Harmonic domain using parametrically deformable models based on (truncated) spherical harmonic expansions

$$f(\theta, \phi) = \sum_{l=0}^{B} \sum_{m=-l}^{l} \hat{f}_l^m Y_l^m(\theta, \phi), \tag{6}$$

where

$$Y_l^m(\theta, \phi) = \sqrt{\frac{(2l+1)(l-m)!}{4\pi(l+m)!}} P_l^m(\cos\theta)e^{jm\phi} \tag{7}$$

and the $\hat{f}_l^m$ are the spherical harmonic coefficients with degree l and order m, B is the maximal bandwidth. The P_l^m are the associated Legendre polynomials, j is the imaginary number. Parametric active surfaces have e.g. been used in [8] and have the advantage that they not only yield smooth surfaces but also a parametric shape description.

Parameterization The first step of the active surface implementation is the choice of an appropriate parameterization. For the segmentation of star-shaped objects, an easy way of parameterizing a shape is describing it in spherical coordinates (θ, ϕ, r) with the two polar variables θ and ϕ, where $0 \leq \theta \leq \pi$ and $0 \leq \phi < 2\pi$ and $r = s(\theta, \phi)$ (compare [9]). r is the Euclidean distance of the surface from the center. For a so parameterized active surface $s(\theta, \phi)$, we can directly compute the corresponding shape descriptor as $\mathbf{d} = (\hat{s}_0^0, \hat{s}_1^{-1}, \hat{s}_1^0, \hat{s}_1^1, \ldots, \hat{s}_l^l)^T$. For a perfect sphere, $s(\theta, \phi) = const.$ and $\hat{s}_l^m = 0$ for all $l, m \neq 0$.

Alternatively, for the more general case of objects with spherical topology, [10] proposes a suitable parameterization that assigns to every pair of angles a three-tuple of coordinates: $\mathbf{s}(\theta, \phi) = (x(\theta, \phi), y(\theta, \phi), z(\theta, \phi))^T$, thus laying a latitude-longitude grid over the shape. This parameterization has been used e.g. in [8] to perform a parametric deconvolution of 3D images. However, for our application, the first and easier parameterization is sufficient because when dealing

with cells, we always expect to find star-shaped objects. This easier parameterization also has the advantage that no explicit internal energy is needed. The regularization can be done implicitly by limiting the bandwidth of the spherical harmonic expansion. This is more difficult when the second parameterization is used: this parameterization tends to artificially introduce sharp edges even at low bandwidth values (compare [8]).

External Forces The external forces $\mathbf{F}_{ext}$ are computed from the filter response $\bar{A}$. This filter response is usually well representing all positions, where the edges are similar to those chosen by the user in the sample data. Due to the normalization, we can even handle linear gray scale variations. On the other hand, the normalization also causes some spot-like filter responses in the background (compare fig. 3d). Accounting for the fact that we are searching for object surfaces, i.e. locally plane-like structures, we can get rid of these wrong filter responses simply by applying the steerable filter for plane detection described in [11]. The filter response is used as edge map. On the gradients of the edge map, we compute the gradient vector flow (GVF) [6] to get a smooth force field . Then we can start with the surface adaptation.

Active Surface Evolution As we are initializing the active surface as a sphere, at the beginning we have $s(\theta, \phi) = const..$. For every iteration, we project the forces that act on the surface onto their radial components, and compute their spherical harmonic expansion. The actual surface update can be performed in spherical harmonic domain. To get the new surface positions in the spatial domain, we then need to perform an inverse spherical harmonic transform.

When the process is finished on the whole database, the user can again interact. If the segmentation is not sufficient for all datasets, the user chooses a new sample and runs through steps 2.1 and 2.2. The new appearance clusters are added to the model and all wrongly segmented datasets of the previous iteration are segmented using this new model.

3   Experiments

The evaluation was performed using three sets of 3D recordings of living tobacco leaf protoplasts (cells lacking the cell wall). Plant protoplasts are a unique tool to study e.g. the function of the plasma membrane, cellular reprogramming and development [12]. An exact segmentation of the cells is needed for various applications, such as the description of the cell anatomy itself or of developmental processes in a meaningful anatomical coordinate system.

3.1   Data

Samples containing one to three single cells were recorded by confocal laser scanning microscopy (CLSM). Single cells were detected using a voter-based Hough transform for spheres [4], which provides us with a good estimate of the object's center $\mathbf{c}$ and the radius r, and cropped to separate volumes. The processed data
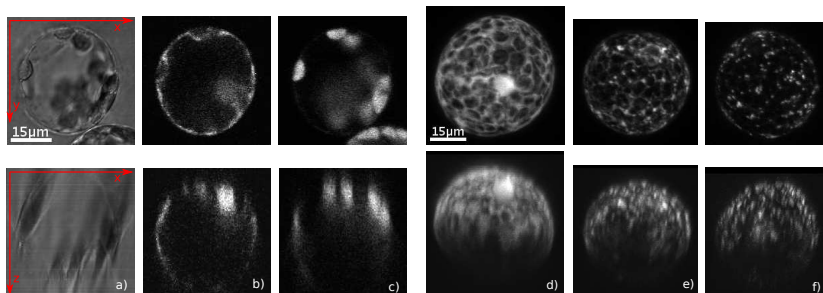
Fig. 2. (a-c) Slices of the recorded channels: a) transmitted light. b) protein pattern (Cyt). c) chloroplasts. (d-e) maximum intensity projections in z-direction (top) and in y-direction (bottom) of the protein stainings: d) Cyt. e) ER. f) GA.

volumes have dimensions ranging from $159 \times 118 \times 71$ voxels for small cells to $509 \times 350 \times 269$ voxels for larger cells. The spatial resolution in xy-direction is $0.28 \times 0.28 \mu m^2$ while the resolution in z-direction is either 0.4 or $0.5\mu m$. The used imaging technique imposes some special image properties, like artefacts from the point spread function (PSF), noise that is generated at the different stages of the signal chain and absorption resulting in signal intensity attenuation in recording direction, i.e. z-direction (see bottom row in fig. 2). The cells have been recorded on three successive days resulting in different cell shapes from roundish over elongated to more complex deformed shapes. Three channels have been recorded, a transmitted light channel and two confocal fluorescence channels, one showing the auto-fluorescence of the chloroplasts and another showing the fluorescence of the tagged protein (see fig. 2a-c). Here we used the protein pattern channel for segmentation. Three different protein patterns have been recorded which all have a different appearance (see fig. 2d-f). While the cytoplasm (Cyt, 55 cells in the database) fills the space between the chloroplasts, the endoplasmatic reticulum (ER, 46 cells) forms a mesh structure and the golgi apparatus (GA, 86 cells) is organized in spot-like structures. All protein patterns are located in the outer shell of the cell, i.e. between outer cell membrane and inner vacuole membrane (cf. fig. 2b).

3.2   Segmentation

The whole segmentation process was performed on slightly smoothed data, we applied a Gaussian smoothing with $\sigma = 0.28\mu$m. For the user specified boundary filtering step, we had to specify certain parameters. The chosen profile length is 8 at a stepsize of 0.56μm which is the double voxel size in xy-direction. The steerable filter, which was used for filtering out spot-like filter responses in the background, also has a parameter σ_s that specifies the thickness of the planes it searches for. We have set σ_s to 0.56μm. Finally, the active surfaces were initialized with the estimated radius r from the detection step. The bandwidth was limited to 16 bands. For the cells with stained Cyt, we have displayed the first
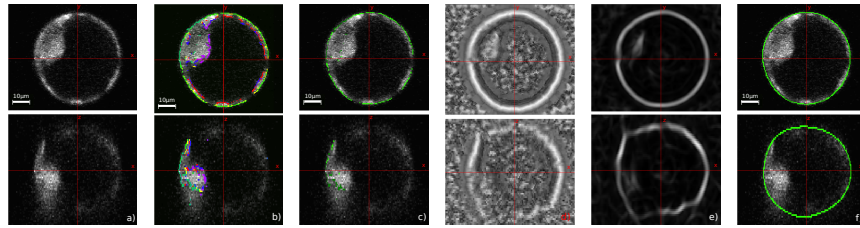
Fig. 3. a) Sample with Cyt staining. b) The clustering result from the K-means clustering with six clusters. Inner and outer boundaries lie, as expected, in different clusters. One can clearly see, that the absorption in z-direction leads to less candidates for edges and thus less training examples for the edge appearance in the lower z-regions. c) The two clusters colored in green were chosen as relevant. d) The filter response $\bar{A}$. e) The used edge map (after the application of the steerable filter). f) The segmentation result.

training sample and the results of the different steps of the presented framework in fig. 3 in two orthogonal views.

3.3   Evaluation and Results

To evaluate our method, we applied the segmentation to all 187 cells in our database. For each of the three patterns, one cell was randomly chosen as first training cell. The segmentation results for all cells were visually inspected in a 3D slice viewer by two experts, who gave label 1 if the segmentation was correct, and label 0 if not. For those cells that were not correctly segmented after the first iteration, a second training step was performed: one of the cells with label 0 was chosen for each pattern as training sample. Altogether, we performed three iterations. The results can be seen in table 1. Most of the cells were already correctly segmented after the first iteration. Some results can be seen in fig. 4.

For further evaluation, we compared the segmentation carried out with our

Experiment	# of cells	Iteration 1	Iteration 2	Iteration 3
Cytoplasm	55	85.5%	**94.6%**	**94.6%**
Golgi	86	86.1%	88.4%	**96.5%**
ER	46	91.3%	95.7%	**97.8%**

Table 1. Results of our method after 1, 2 and 3 iterations.

method to the results that could be achieved with the ITK segmentation tool ITK-SNAP [3], which is based on 3D geodesic active contours. The internal forces are based on the gradient magnitude in the dataset. The preprocessing as well as the active contour parameters have to be manually adjusted for each dataset, which is why we performed this segmentation only for three datasets.
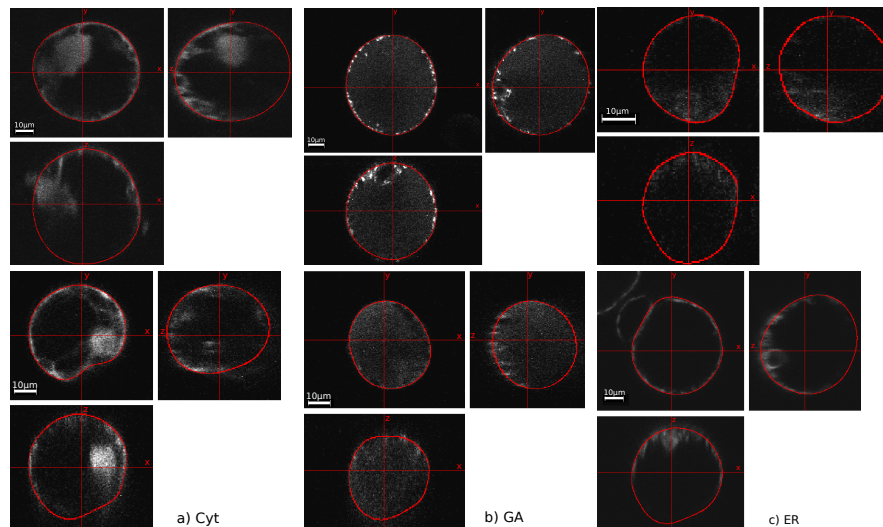
Fig. 4. Segmentation results in three orthogonal views. We have displayed two examples for every stained pattern.

The ITK-SNAP preprocessig parameters are: the scale of the Gaussian blurring σ, which we set to 0.56μm, the edge contrast κ, which we set to 0.1 and the edge mapping exponent, which we set to 1.6. For the geodesic active surfaces, some parameters have to be adjusted as well. Here, we could not use the same parameters for all three cells. We manually initialized the contours from outside, because the internal structures of the cells made an initialization from inside impossible. The balloon force was set to a value between -0.7 and -0.9, the curvature force, that ranges from 0 (detailed) to 1 (spherical) was set to 0.8 and the advection force, that pushes the boundary back as it tries to cross edges, was set to 4.0. Additionally, the iteration has to be stopped manually. We needed between 461 and 791 steps to get the results displayed in fig. 5. As it can be seen, this segmentation tool can, despite all manual interaction, not handle the heterogeneous boundary and the absorption in z-direction.

4   Conclusion

We have presented a semi-supervised segmentation method for volumetric datasets, that can handle heterogeneous edge appearances. In our framework, edge models are learned from user input, while the user interaction is limited to very simple and intuitive operations and no further low-level parameters have to be adjusted. Although current 3D confocal recordings of single plant cells pose numerous challenges, such as inhomogeneous object boundaries, strong gray-value attenuations and noise, the segmentation of the datasets using the proposed method resulted in a highly reliable identification of cell boundaries.
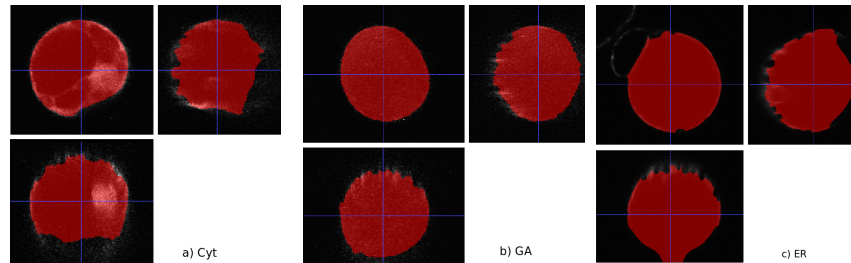
Fig. 5. Segmentation results with ITK-SNAP for the same cells as in fig. 4, second row.

5   Acknowledgments

References

1. Cootes, T., Edwards, G., Taylor, C.: Active appearance models. IEEE Trans. on PAMI **23/6** (2001) 681–685
2. Vehkomäki, T., Gerig, G., Szkely, G.: A user-guided tool for efficient segmentation of medical image data. LNCS, Springer, vol. 1205 (1997) 685–694
3. Yushkevich, P.A., Piven, J., Hazlett, C., H., S., Smith, G., Ho, R., Ho, S., Gee, J.C., Gerig, G.: User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability. Neuroimage **31/3** (2006)
4. Schulz, J., Schmidt, T., Ronneberger, O., Burkhardt, H., Pasternak, T., Dovzhenko, A., Palme, K.: Fast scalar and vectorial grayscale based invariant features for 3d cell nuclei localization and classification, LNCS, Springer, vol. 4174 (2006)
5. Xu, R., Wunsch, D.C.: Clustering. Wiley (2008)
6. Xu, C., Prince, J.: Snakes, shapes, and gradient vector flow. IEEE Trans. Imag. Proc. **7/3** (1998) 321–345
7. Keuper, M., Padeken, J., Heun, P., Burkhardt, H., Ronneberger, O.: A 3d active surface model for the accurate segmentation of *drosophila* schneider cell nuclei and nucleoli, LNCS, Springer, vol. 5875 (2009) 865–874
8. Khairy, K., Howard, J.: Spherical harmonics-based parametric deconvolution of 3d surface images using bending energy minimization. Medical Image Analysis **12** (2008) 217–227
9. Ballard, D.H., Brown, C.M.: Computer vision. Prentice-Hall, NJ (1981)
10. Brechbühler, C., Gerig, G., Kübler, O.: Parametrization of closed surfaces for 3-d shape description. Comput. Vis. Image Underst. **61**(2) (1995) 154–170
11. Aguet, F., Jacob, M., Unser, M.: Three-dimensional feature detection using optimal steerable filters. In: Proc. of the ICIP. (2005) 1158–61
12. Dovzhenko, A., Bergen, U., Koop, H.U.: Thin alginate layer (tal)-technique for protoplast culture of tobacco leaf protoplasts: Shoot formation in less than two weeks. Protoplasma **204** (1998) 114–118