Regularized Geometric Hulls for Bio-medical Image Segmentation

Marco Körner, Mahesh V. Krishna, Herbert Süße, Wolfang Ortmann, and Joachim Denzler

Computer Vision Group,
Friedrich Schiller University of Jena, Ernst-Abbe-Platz 2, 07745 Jena, Germany
⟨marco.koerner@uni-jena.de⟩

Abstract

One of the most important and challenging tasks in bio-medical image analysis is the localization, identification, and discrimination of salient objects or structures. While to date human experts are performing these tasks manually at the expense of time and reliability, methods for automation of these processes are evidently called for. This paper outlines a novel technique for geometric clustering of related object evidence called *regularized geometric hulls (RGH)* and gives three exemplary real-world application scenarios. Several experiments performed on real-world data highlight a set of useful advantages, such as robustness, reliability, as well as efficient runtime behavior.

1 Introduction

In bio-medical image processing, spatial and semantic clustering of salient objects is a fundamental part of the image segmentation pipeline, *e.g.*, when meaningful structures have to be identified from microscopic images. As performing this task manually by medical experts is slow, costly, and prone to errors, the need for automatic methods is obvious. While methods like *active contours* or *snakes* [Kass et al., 1988] and all their modifications are commonly used to tackle these problems, their iterative nature might inhibit their usage in real-time scenarios. Moreover, they are found on a continuous energy minimization problem formulation which has to be discretized in order to apply them to digital image data. Above all, they also usually involve a large set of parameters which are difficult to interpret and adjust.

In contrast, *regularized geometric hulls (RGH)* [Körner et al., 2014] are designed to directly operate on discrete data and solely rely on one single parameter to be tuned easily by the user. In this paper, after reviewing this concept, we will further demonstrate the practical use of RGHs and their advantages by means of three real-world application scenarios.

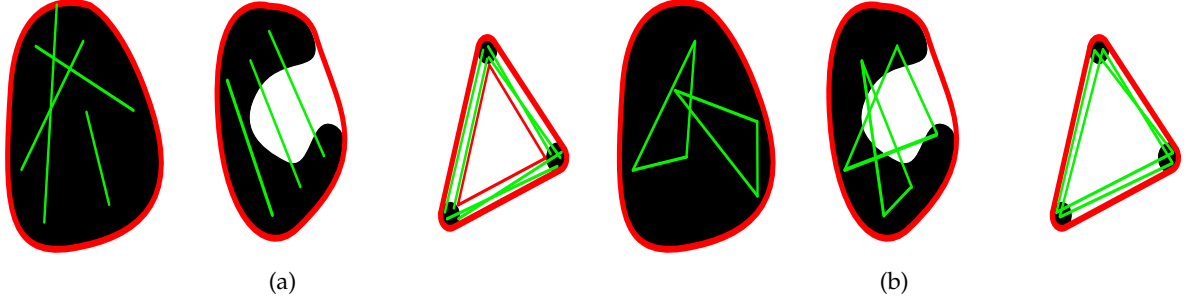(a)   (b)

Figure 1: Given a set of data points (black), a convex set (red) encloses this population in a way that all points along a straight line connecting two distinct data points themselves are elements of this convex set. Canonically, all points within a triangle of data points are elements of this convex set. As can be seen from these examples, the classical convex hull does not preserve holes or unconnected components represented by the data.

2   Regularized Geometric Hulls

This section is dedicated to briefly introduce into the concept of *regularized geometric hulls (RGH)* recently presented in [Süße et al., 2014, Körner et al., 2014], which is the core ingredient of our proposed methods. These can be regarded as an extension of classical *convex hulls*:

Definition 1 (CONVEX SET).
   A subset $\mathcal{S} = \{s_i\} \subseteq \mathbb{V}^D$ from any D-dimensional vector space $\mathbb{V}^D$ is called convex, *if for any pair $(s_i, s_j) \in \mathbb{V} \times \mathbb{V}$*

$$(1 - \lambda)s_i + \lambda s_j \in \mathcal{S} \quad \forall 0 \leq \lambda \leq 1 \quad . \tag{1}$$

Definition 2 (CONVEX HULL).
   Given an arbitrary set of Euclidean points $\mathcal{P} = \{p | p \in \mathbb{R}^D\}$ (or discrete lattice points $\mathcal{Q} = \{q | q \in \mathbb{Z}^D\}$), the convex hull $\mathcal{P}_H \supseteq \mathcal{P}$ ($\mathcal{Q}_H \supseteq \mathcal{Q}$) *is the smallest convex subset of $\mathbb{R}^D$ that contains $\mathcal{P}$ ($\mathcal{Q}$).*

From Def. 1 one can safely derive the fact that for any triple $(s_i, s_j, s_k) \in \mathbb{V} \times \mathbb{V} \times \mathbb{V}$ of convex set elements, their linear combination $(1 - \lambda - \nu)s_i + \lambda s_j + \nu s_k \in \mathcal{S}, \forall 0 \leq \lambda, \nu \leq 1$, is also an element of this set. Hence, each triangle induced by elements from such a convex set is ensured to be entirely covered by this convex set, as illustrated in Fig. 1.

While convex hulls provide geometric descriptions of objects—*i.e.*, discrete point sets or contours—, and various algorithms for their computation with fair runtime complexity $\mathcal{O}(n \log n)$ exist in the field of computational geometry, they do not preserve holes in the data point population or unconnected object components. As this convex hull approximation of points thus is insufficiently coarse for most computer vision problems, we aim to overcome this problem and further allow for a certain degree of non-convexity—*i.e.*, concavity—while computing geometric hulls by defining the concept of *regularized geometric hulls (RGH)*:

Definition 3 (ADJACENT POINTS).
   Let d : $\mathcal{M}^D \mapsto \mathbb{R}$ *be a metric in* $\mathcal{M}^D$. *Two points* $p, q \in \mathcal{M}^D$ *are called ζ-adjacent wrt. a non-negative constant $0 \leq \zeta \in \mathbb{R}$, if*

$$p \sim^{\zeta} q \Leftrightarrow \mathrm{d}(p, q) \leq \zeta \quad . \tag{2}$$

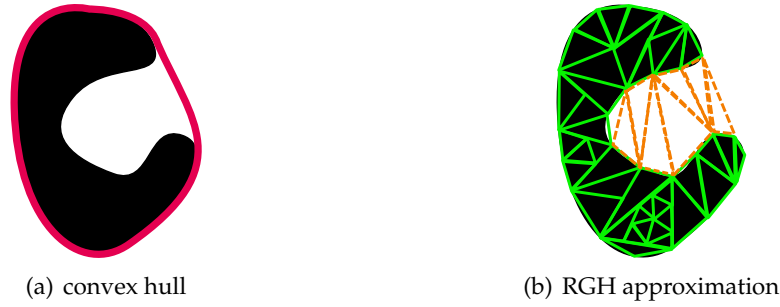(a) convex hull   (b) RGH approximation

Figure 2: While the (a) classical convex hull (solid red curve) is defined as the union set of all lines connecting all pairwise point pairs from the input data, (b) the regularized geometric hull approximates the data population by a set of triangles (solid orange) excluding those with edges exceeding a certain maximal length (dashed orange). By doing so, relevant properties of the original data—*e.g.*, strong concavities, unconnected components, and holes— are preserved.

Definition 4 (REGULARIZED GEOMETRIC HULL (RGH) FOR EUCLIDEAN POINT SETS)**.**
Let $\mathcal{P} = \left\{ \boldsymbol{p} | \boldsymbol{p} \in \mathbb{R}^D \right\}$ be an arbitrary set of Euclidean points and $\Delta(\boldsymbol{p}_1, \boldsymbol{p}_2, \boldsymbol{p}_3) \subseteq \mathcal{P}$ the set of all points $\boldsymbol{p}_i \in \mathcal{P}$ enclosed by the triangle defined by the 3-tuple $(\boldsymbol{p}_1, \boldsymbol{p}_2, \boldsymbol{p}_3) \in \mathcal{P}^3$. Further, let

$$\mathcal{P}_\Delta^\zeta = \left\{ \Delta(\boldsymbol{p}_1, \boldsymbol{p}_2, \boldsymbol{p}_3) \subseteq \mathcal{P} \mid \boldsymbol{p}_1 \sim^\zeta \boldsymbol{p}_2 \wedge \boldsymbol{p}_2 \sim^\zeta \boldsymbol{p}_3 \wedge \boldsymbol{p}_1 \sim^\zeta \boldsymbol{p}_3 \right\}, \quad 0 \le \zeta \in \mathbb{R}, \qquad (3)$$

be the set of all triangles of ζ-adjacent points in $\mathcal{P}$. Then, the union set

$$\mathcal{P}_H^\zeta = \bigcup \mathcal{P}_\Delta^\zeta \cup \mathcal{P} \qquad (4)$$

is called the Regularized Geometric Hull (RGH) *of $\mathcal{P}$ wrt. ζ.*

According to these definitions, the RGH approximates the input data population by a union set of triangles induced by triples of data points, while triangles with edge lengths larger than ζ are omitted, as illustrated in Fig. 2. It can be clearly seen that the properties of the geometric hull $\mathcal{P}_H^\zeta$ strictly depend on the particular choice of the structure parameter ζ, as illustrated in Fig. 3. If $\zeta = 0$, the geometric hull $\mathcal{P}_H^0$ is identical to the point set $\mathcal{P}$. In turn, if $\zeta \to \infty$, the geometric hull $\mathcal{P}_H^\zeta$ converges to the convex hull $\mathcal{P}_H$ yielding the relation

$$\mathcal{P} = \mathcal{P}_H^0 \subseteq \mathcal{P}_H^\zeta \subseteq \mathcal{P}_H^\infty = \mathcal{P}_H, \quad 0 \le \zeta \in \mathbb{R}. \qquad (5)$$

Hence, as illustrated in Fig. 3, the structure parameter ζ regularizes the convexity or concavity of the geometric hull. In contrast to the classical convex hull $\mathcal{P}_H$, the RGH $\mathcal{P}_H^\zeta$ describes the geometric structure of the input data $\mathcal{P}$ by a set of outer and inner contours.

2.1 Approximation

Def. 4 suggest that the naïve implementation to compute the RGH shows cubic complexity $\mathcal{O}(n^3)$ in n the number of input points which makes its application impractical in realistic scenarios. For this reason, we approximate the underlying set of triangles $\mathcal{P}_\Delta^\zeta \subseteq \mathcal{P}^3$ by a
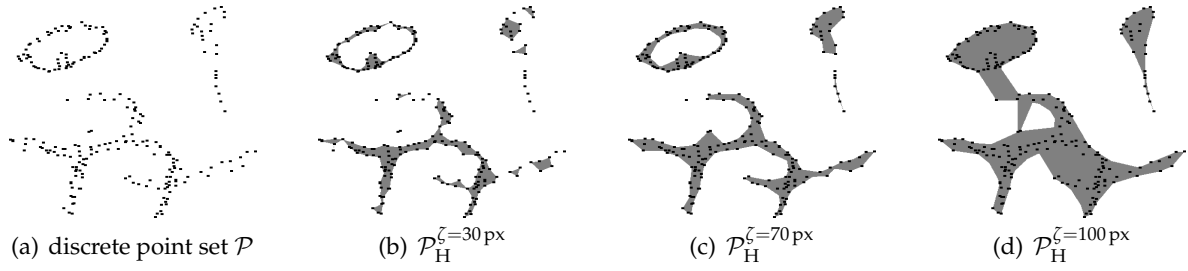
(a) discrete point set $\mathcal{P}$ (b) $\mathcal{P}_H^{\zeta=30\,\text{px}}$ (c) $\mathcal{P}_H^{\zeta=70\,\text{px}}$ (d) $\mathcal{P}_H^{\zeta=100\,\text{px}}$

Figure 3: Influence of different values for the regularization parameter ζ on RGH estimated from dense point sets. Note that RGHs preserve inner contours and unconnected components.

(a) original discrete point set $\mathcal{P}$ (b) outer contour smoothed by RGH   (c) Delaunay-like approximation of RGH   (d) result of morphological closing
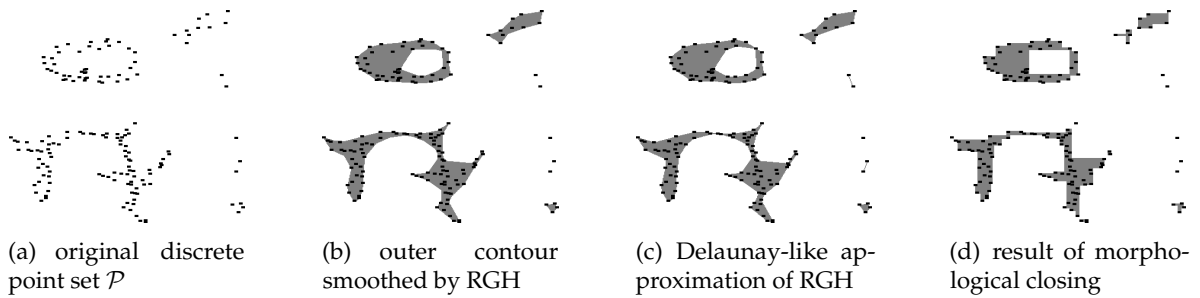
Figure 4: *RGH*s can be approximated by Delaunay-like triangulations of the input point cloud $\mathcal{P}$ realizing a morphological closing operation with adaptive, polygonial structure elements.

modification of the *Delaunay triangulation* [Delaunay, 1934] $\mathcal{P}_\triangle^{\zeta_{\text{DT}}} \subseteq \mathcal{P}^3$, which is supposed to be a triangular tessellation of a given point set $\mathcal{P}$ optimal wrt. a given distance criterion. All triangles obtained by Delaunay's triangulation algorithm are drawn into a binary image, wherefrom final contours are extracted afterwards. Hence, the Delaunay triangulation $\mathcal{P}_\triangle^{\zeta_{\text{DT}}}$— restricted to triangles with edge lengths smaller than $0 \leq \zeta \in \mathbb{R}$—can be interpreted as an approximation of the previously used set $\mathcal{P}_\triangle^{\zeta}$ of triangles of ζ-adjacent points. Fig. 4(c) shows the results of the RGH algorithm using the Delaunay-like approximation of a discrete point set. Compared to the result of the original RGH algorithm shown in Fig. 4(b), the results are reasonably good and useful for further processing. As the Delaunay algorithm shows quasi-linear complexity $\mathcal{O}(n \log n)$ in n the number of vertices, this approximation remarkably speeds up the whole computation and allows for real-time performance and scales linearly with number of the input data points in both the 2d and the 3d case.

2.2 Geometric Interpretation

When analyzing the results of the RGH or their approximation based on Delaunay triangulation, one can observe various handy properties useful for further processing of precomputed contours or point sets. First, the RGH augments the input contour or dense point set $\mathcal{P}$ by a *geometric orientation*, enabling us to easily apply neighborhood-based methods for shape recognition. While doing so, in the case of discrete point sets and dependent from the structure parameter ζ, existing holes are preserved and isolated objects are detected, as can be seen in Fig. 4(b) and (c).

Second and as illustrated in Fig. 4, the presented method performs a *smoothing* of a given
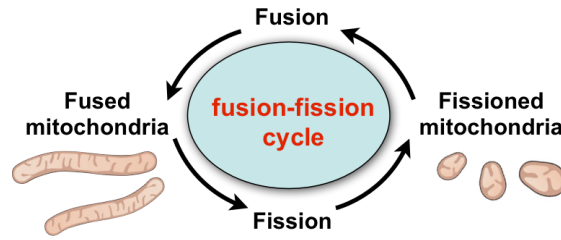
Figure 5: According to their state within the fusion-fission cycle, mitochondria appear either line-shaped or fragmented. In pathological conditions, mitochondria remain longer in the fissioned stage or show other types of variations, which can be observed by image processing.

contour which can be used to deal with inaccurate segmentation. Again, the impact of this smoothing is tuned by the structure parameter ζ. In terms of computational geometry, this is comparable to the *morphological closing* operator applied to binary objects. So, RGHs can be characterized as such a closing operation based on adaptive, polygonal structure elements. In contrast to classical morphological closing with constant regular structure elements (*e.g.*, rectangles, ellipses), the results of RGHs constructed from contours or dense point sets appear much more smooth and intuitive, as can be seen in Fig. 4(d).

3 Application Scenarios

As indicated in the previous section, RGHs can be used for geometric clustering of generic data populations in Euclidean space. In order to emphasize their wide applicability, we will present three different exemplary real-world application scenarios in this section.

3.1 Boundary Estimation of Neural Cells in Confocal Laser Scanning Microscopy Images

Mitochondria are sub-cellular organelles that have the important function to provide energy production for the cell. Furthermore, they contribute to intracellular calcium handling, free-radical scavenging and induction of apoptosis. Live cell microscopy studies showed that mitochondria are highly dynamic organelles that are organized as large interconnected networks with dynamic fusion and division processes (*cf.* Fig. 5). Their lengths and the degree to which they form closed networks are determined by the balance between fission and fusion rates. This fine-tuned balance is influenced by metabolic as well as pathologic conditions in mitochondria and their cellular environment. Alterations in mitochondrial size and shape and also metabolic dysfunctions of mitochondria are found in neurodegenerative diseases [Chan, 2006, Westermann, 2010] as well as *diabetes mellitus* and cancer. Additionally, mitochondrial fission and fusion play prominent roles in disease-related processes such as apoptosis and mitophagy. The quantitative analysis of *mitochondrial fragmentation* in whole cells is difficult and very error-prone due to a high density of mitochondria in the cell body—as can be seen by the example given in Fig. 6(a)—but nevertheless would provide the basis for an evidence-based description of mitochondrial dysfunction as well as for the evaluation of mitochondrial therapeutic interventions. Therefore, in order to provide methods for quantitative analysis of mitochondrial fragmentation in *confocal laser scanning microscopy (CLSM)*, we used RGHs for clustering detected mitochondrial structures enabling for volumetric analysis. The algorithm
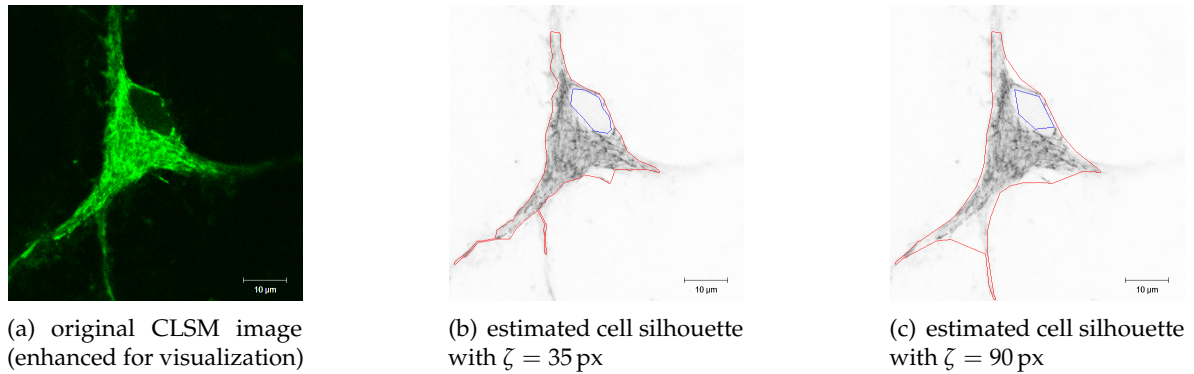
(a) original CLSM image
(enhanced for visualization)

(b) estimated cell silhouette
with $\zeta = 35\,\text{px}$

(c) estimated cell silhouette
with $\zeta = 90\,\text{px}$

Figure 6: From (a) a typical z stack slice of a neural cell as input, our algorithm can determine the boundaries of the cell body (red) and the nucleus (blue) based on the RGH computation performed on the segmented structure.

(a) original CLSM image
(enhanced for visualization)

(b) midlines of detected mitochondria fragments
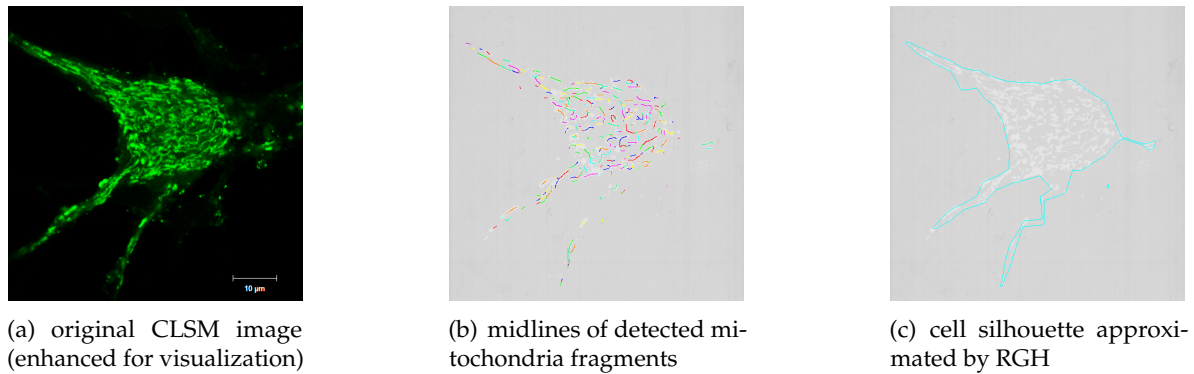
(c) cell silhouette approximated by RGH

Figure 7: Intermediate results of individual stages of our framework for cell boundary estimation from CLSM images.

allows for the semi-automatic analysis of mitochondrial fragments in whole cell z stacks, thus avoiding random sampling of single mitochondria and gaining maximal objective results due to minimal user interaction.

As can be seen in Fig. 6(a), CLSM images show a high density of mitochondria and are further affected by pixel noise, which handicaps the automatic segmentation of mitochondrial structures. In order to overcome this problem, we take advantage of the *oriented differences of boxes (ODoB)* operator presented for segmenting blood vessels in *confocal laser endomicroscopy (CLE)* images [Süße et al., 2013]. For segmenting mitochondrial structures, we use local maxima of precomputed ODoB outputs to start instances of *seeded region growing (SRG)*, which will result in a binary segmentation map distinguishing between hypothetical mitochondria pixels and the background. This completely unrelated set of foreground pixels is used to create a morphological skeleton, wherefrom in turn a forest of *minimum spanning trees (MST)* is extracted in order to augment the initial set of unrelated foreground points with a neighborhood relation. In order to overcome the problem of over-segmentation, small artifacts are eliminated by applying a *split-and-merge (SAM)* strategy. One major observation for pathological cases is the amount of mitochondrial fragmentation within the cell body.
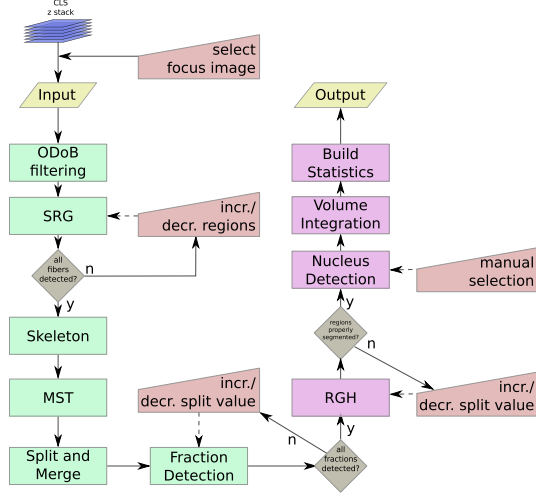
Figure 8: The outline of our framework: dashed and solid lines attached to trapezoids indicate optional and mandatory user interaction, respectively.

Table 1: Statistics computed from the segmented and integrated z stack slices reflecting structural and pathological properties of the mitochondria population within cytoplasm.

Parameter	Symbol	Expression
no. of mitoch. fragments	n	
avg. mitoch. fragm. length	μ_{mito}	
mitoch. fragm. len. stddev.	σ_{mito}	
fractal dimension	D	
lacunarity	Λ	
cell body area	A_{cell}	
nucleus area	A_{nucleus}	
2D mitoch. density	ρ_{2D}	$= \frac{n}{A_{\mathrm{cell}} - A_{\mathrm{nucleus}}}$
cell body volume	V_{cell}	$= h_z \cdot \sum A_{\mathrm{cell}}$
nucleus volume	V_{nucleus}	$= h_z \cdot \sum A_{\mathrm{nucleus}}$
3D mitoch. density	ρ_{3D}	$= \frac{\sum n}{V_{\mathrm{cell}} - V_{\mathrm{nucleus}}}$

In order to locate fragmentation positions along the mitochondria center line, we employ a sliding window approach traversing the segmented path. For each position, we identify local minima by applying a top-hat filter, which produces evident outputs at extrema positions (*cf.* Fig. 7(b)). A more detailed description of these methods can be found in [Süße et al., 2014].

We now have access to an accurate segmentation of line structures within the cell which are considered to be fused mitochondria. As mitochondrial structures are equally distributed among the cell body, we use the these structures as input for RGH computation in order to estimate the actual cell body contour as well as the nucleus. Fig. 6 shows that the outer contour encloses the whole cell body, while the smoothness depends on the choice of the structure parameter ζ. Furthermore, as the RGH algorithm is able to deal with nested contours (*i.e.*, holes), the cell nucleus is clearly segmented and can be ignored for further analysis.

After integrating these results over a series of z stacks, various statistical parameters are computed excluding the cell nucleus volume, as outlined in Fig. 8 and Tab. 1.

Experiments   In order to evaluate the proposed method, we acquired data of stained mitochondria using a ZEISS LSM 710 *confocal laser scanning microscope (CLSM)*, as exemplary shown in Fig. 6(a) and Fig. 7(a). A more detailed description of the image acquisition procedure is given in [Süße et al., 2013].

As we are not aware of any standard procedure to compare manual and automatic segmentation of mitochondria, we present a collection of qualitative examples. In Fig. 9, one exemplary result is shown in comparison to a manual segmentation performed by an expert. Referring also to Fig. 10, it can be seen that the segmentation results obtained by our proposed method approximate the expert segmentation in a reasonable good way. This observation was approved by several medical experts. As mentioned before, the segmentation of mitochondria is used for computing statistical parameters in order to quantify the degree of fragmentation. One very important feature is the distribution of mitochondria segment lengths, which is exemplary shown in Fig. 9(c). To date, mitochondria analysis is based on manual length

(a) automatically seg-
mented mitochondrial
structures

(b) manually seg-
mented mitochondrial
structures

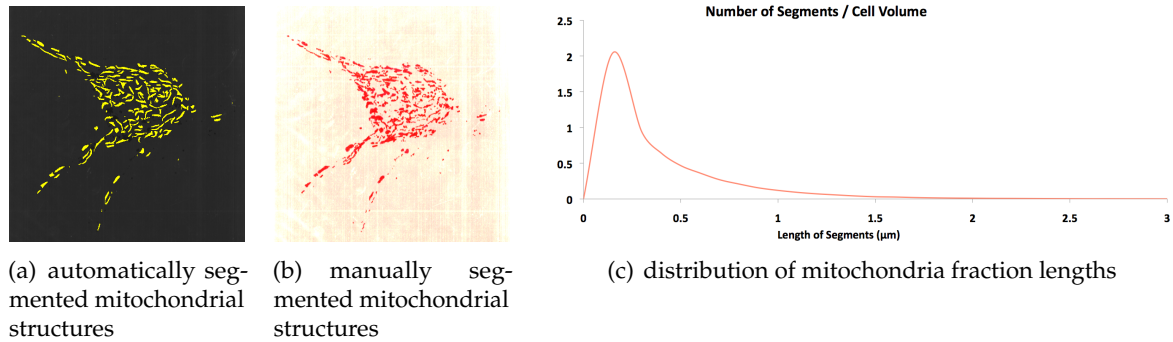(c) distribution of mitochondria fraction lengths

Figure 9: Qualitative comparison between manual and automatic segmentation of mitochondrial structures in neural cells and derived fraction length statistics.
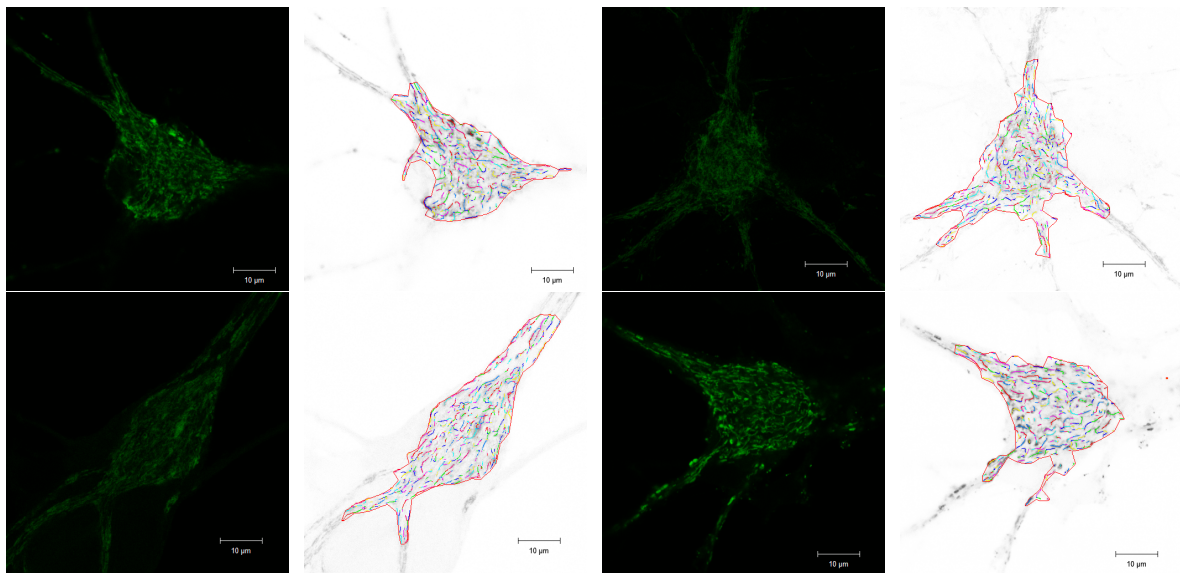

Figure 10: Examplary qualitative results of our proposed mitochondria detection method.

measurements of randomly selected individual mitochondria, which is very time-consuming and error-prone. In contrast, our approach operates instantly and so speeds up the whole diagnostic pipeline in a remarkable extend.

3.2   Cell Nucleus Segmentation in Fluorescence Microscopy Images

For microscopy-based tissue analysis, the identification and segmentation of individual cells is one of the most important and fundamental steps within the entire processing pipeline. While classically this task had to be performed manually, more and more automatic or semi-automatic methods are being developed. As this problem is highly affected by image degradations such as blur or pixel noise, traditional contour segmentation methods are likely to fail in these scenarios. Furthermore, other approaches require manual annotation of cells for localization or employ computationally expensive machine learning techniques for building complex models of cell appearance. Segmentation of cells from images has been tackled with a wide variety of tools, ranging from simple thresholding to machine learning-based

(a) microscopic image (enhanced for visualization) (b) corresponding nuclei expressing H2B-mCherry (c) classified nuclei pixels (d) RGH result
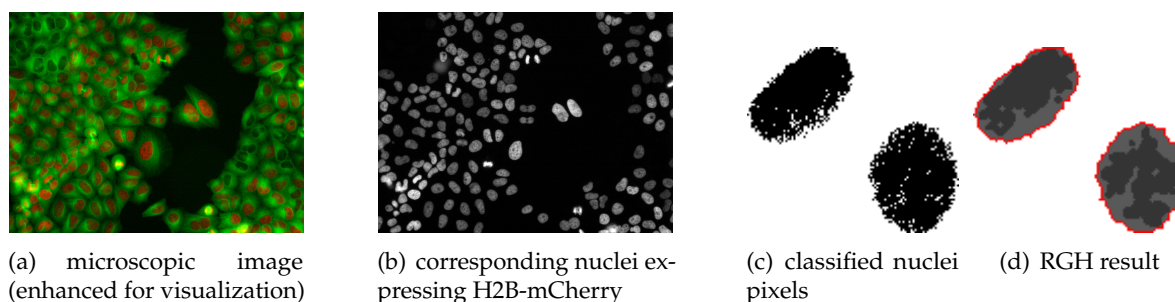
Figure 11: For detection cell nuclei in (a)+(b) fluorescence microscopy images from the CELLCOGNITION dataset are (c) initially thresholded and further (d) used to identify nuclei boundary regions by RGH.

methods [Meijering, 2012]. Many of them target specific types of microscopy images and are thus application-specific. Other approaches based on *seeded watershed transform* [Meyer and Beucher, 1990] suffer from oversegmentation problems and the quality of the results depends on the initial seeds.

Arteta *et al*. employ a SVM classifier for categorizing each pixel to belong to either cell foreground or background [Arteta et al., 2012]. This binarization is further analyzed through hierarchical graph construction from *maximally extremely stable regions (MSER)* to achieve final segmentation results. While this approach shows promising results, necessity of labeled training data and pixel-wise operations on high-dimensional image sequences make it unsuitable for fast processing.

In contrast, we show how to apply RGH for fast and robust joint cell detection and segmentation. Once one rather simple parameter is set, the method runs fully automatically. As the proposed method is designed in a very general way, the overall framework is rather straightforward. Following the procedure outlined in [Körner et al., 2014], after binary thresholding the input images (*cf*. Fig. 11(c)), they are directly passed to the RGH module to create foreground-background images as exemplary shown in Fig. 11(d). The values of the structure parameter ζ are varied interactively on-the-fly in order to show its influence on the overall results. Cells are finally extracted by subsequent contour extraction.

Experiments   As annotated image data of cells is very rare, most of the existing approaches were evaluated on rather small non-public databases. For instance, Arteta *et al*. used three datasets of 12 to 22 images in their experiments [Arteta et al., 2012]. In contrast, the CELLCOGNITION dataset presented in [Held et al., 2010] consists of 7 sequences of 206 frames of size 1392×1040 px, each showing RNAi treated human HeLa Kyoto cells expressing fluorescent H2B-mCherry (orange) and α-tubulin (green). In total, the dataset contains 363,120 annotated cell objects with automatically generated ground truth data providing centroids and bounding boxes. Fig. 11 shows an example frame of the dataset, with the full recording and the segregated nuclei. Apart from the huge number of objects, the dataset is made more challenging by low contrast in many cases, and overlapping cells due to limits in the thinness of slide preparations.

For comparison to other methods, we used two popular quantification methods. First, as suggested in [Arteta et al., 2012], we assumed each cell detected by our system to be a
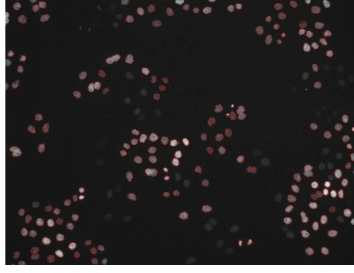
(a) dense population
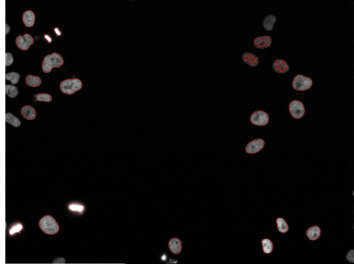
(b) sparse population

Figure 12: The results of the RGH clustering for (a) dense and (b) sparse fluorescence microscopy images.

Table 2: Evaluation of our approach for cell nucleus segmentation from fluorescence microscopy images and comparison to the state-of-the-art.

Method	Accuracy		Runtime	
	centroid distance	pascal criterion	training	testing
	$\rho_{\text{dist}} = 20\,\text{px}$	$\rho_{\text{area}} = 0.5$		
our approach			—	$1.96\,\text{s}/\text{f}$
$\zeta = 3\,\text{px}$	91.65%	85.75%		
$\zeta = 4\,\text{px}$	91.75%	85.01%		
$\zeta = 5\,\text{px}$	92.11%	85.77%		
$\zeta = 6\,\text{px}$	92.02%	85.23%		
[Arteta et al., 2012]	88.79%	86.20%	$6232\,\text{s}$	$130.66\,\text{s}/\text{f}$

true positive, if the distance between its centroid and the centroid of the nearest ground truth cell is smaller than a predefined threshold ρ_{dist}. Usually, this threshold is set to the radius of the smallest cell expected to appear in the data. Second and more commonly used in object detection, a detected cell is assumed to match a ground truth cell, if the ratio

$$r\left(B_{\text{det}}, B_{\text{gt}}\right) = \frac{\text{area}\left(B_{\text{det}} \cap B_{\text{gt}}\right)}{\text{area}\left(B_{\text{det}} \cup B_{\text{gt}}\right)} \tag{6}$$

of the union and intersection areas of respective bounding boxes B_{det} and B_{gt} exceeds the value of $\rho_{\text{area}} = 0.5$ (*cf. PASCAL criterion* [Everingham et al., 2010]).

In order to evaluate the performance of our methods, we compared to the state-of-the-art method presented in [Arteta et al., 2012]. From the numbers reported in Tab. 2, it can be observed that our approach performs better or at least comparable to the competing method. Intuitively, the structure parameter ζ only slightly influences the overall accuracy. Wrong segmentation mainly occurs when two objects are closer than the structure parameter ζ. Furthermore it has to be noted that the evaluation based on cell centroid distances yields higher accuracy. This should be taken into account for further comparisons. Fig. 12 shows some typical results of RGH segmentation for fluorescence microscopy images.

The most remarkable observation arises from the run-time analysis. While the supervised method of [Arteta et al., 2012] requires a very time-consuming training phase, even the testing step is rather slow. In contrast, our approach performs localization and segmentation almost two orders of magnitudes faster, which is a great advantage in real-world applications. Errors mainly occur due to corrupted segmentation.

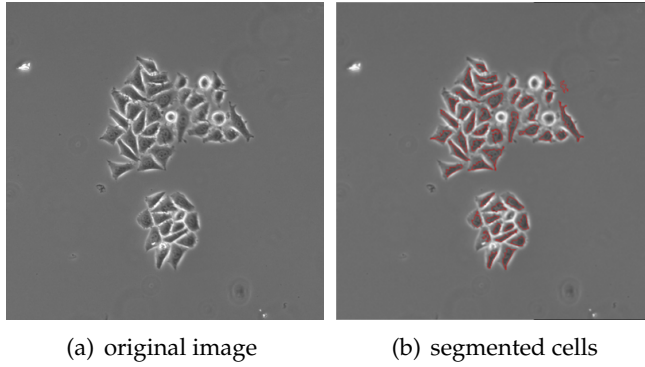(a) original image   (b) segmented cells

Figure 13: Exemplary results for cell segmentation from phase-contrast microscopy images.

Table 3: Evaluation of our approach for cell segmentation in phase-contrast images and comparison to the state-of-the-art.

Method	Accuracy	
	Precision	Recall
our approach		
$\zeta = 3\,\mathrm{px}$	84.38%	86.02%
$\zeta = 4\,\mathrm{px}$	83.15%	82.16%
$\zeta = 5\,\mathrm{px}$	77.72%	69.02%
[Arteta et al., 2012]	90.12%	85.10%

3.3 Cell Segmentation in Phase-Contrast Microscopy Images

In phase-contrast microscopy images, it is common to find densely packed cells with gray interiors, with brightly illuminated borders between them. This might pose a unique challenge to segmentation algorithms, as this small separation between the cells makes it hard to separate them well.

However, we show that similar to the case of fluorescence microscopy, phase-contrast microscopy images can be segmented through the RGH algorithm, without any significant changes. Here, given phase-contrast microscopy images such as the one shown in Fig. 13(a), we follow a similar pipeline—*i.e.*, binarization, application of RGH algorithm and contour extraction—to achieve final segmentation.

Experiments We use the phase-contrast microscopy image dataset supplied with [Arteta et al., 2012], consisting of 11 testing images. The images are of size $400 \times 400\,\mathrm{px}$, each containing 100 to 130 cells on average. As we do not require any learning step, we discarded the training images. The ground truth information provided with the dataset consists of the centroid locations. For evaluation, we use the centroid detection method described in the case of the fluorescence microscopy in order to compare to the original results presented by [Arteta et al., 2012].

The qualitative results of this procedure can be seen in Fig. 13(b). The cells are segmented efficiently and the boundaries conform well to the actual cell boundaries. Tab. 3 shows the quantative results. In comaprison, we have reproduced the results reported in [Arteta et al., 2012] at the closest recall value. Apparently, the RGH-based method achieved comparable results in terms of accuracy and—as pointed out earlier—is much faster than the method of [Arteta et al., 2012] relying on a more sophisticated appearance model.

4   Summary

We introduced the concept of *regularized geometric hulls (RGH)* for data clustering and motivated their practical use by presenting three different applications from the domain of bio-medical image segmentation. We outlined advantages of this tool, for instance their purely discrete formulation, the non-iterative fashion, as well as the efficient runtime realized

by appropriate approximation. Subjected to future work, we intend to predict the structure parameter ζ adaptively during segmentation or learn it from training data and to use more sophisticated preprocessing of the input data and segmentation.

Acknowledgments

References

C. Arteta, V. Lempitsky, J. Noble, and A. Zisserman. Learning to detect cells using non-overlapping extremal regions. In *MICCAI*, pages 348–356. 2012.

DC Chan. Mitochondria: dynamic organelles in disease, aging, and development. *Cell*, 125 (7):1241–1252, 2006.

B. Delaunay. Sur la sphère vide. *USSR Academy of Sciences*, (6):793–800, 1934.

M. Everingham, L van Gool, C. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88(2):303–338, 2010.

M. Held, M. Schmitz, B. Fischer, T. Walter, B. Neumann, M. Olma, M. Peter, J. Ellenberg, and D. Gerlich. Cellcognition: Time-resolved phenotype annotation in high-throughput live cell imaging. *Nature Methods*, 7(9):747–754, 2010.

M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *IJCV*, 1(4):321–331, 1988.

M. Körner, M. Venkata Krishna, H. Süße, W. Ortmann, and J. Denzler. Regularized geometric hulls for robust cell nucleus segmentation from microscopy images. In *MIUA*, pages 117–122, 2014.

E. Meijering. Cell segmentation: 50 years down the road. *Signal Processing Magazine, IEEE*, 29 (5):140–145, 2012.

F. Meyer and S. Beucher. Morphological segmentation. *JVCIR*, 1(1):21–46, 1990.

H. Süße, W. Ortmann, C. Lautenschläger, M. Körner, C. Schmidt, A. Stallmach, and J. Denzler. Oriented differences of boxes operators for blood vessel segmentation and analysis in confocal laser endomicroscopy images with minimal user interaction. In *MIUA*, 2013. 164–169.

H. Süße, W. Ortmann, J. Lautenschläger, C. Lautenschläger, M. Körner, J. Großkreutz, and J. Denzler. Quantitative analysis of pathological mitochondrial morphology in neuronal cells in confocal laser scanning microscopy images. In *IWBBIO*, pages 1290–1301, 2014.

B. Westermann. Mitochondrial fusion and fission in cell life and death. *Nature reviews Molecular cell biology*, 11(12):872–884, 2010.