Image Segmentation using Snakes and Stochastic Watershed

With Applications to Microscopy Images of Biological Tissue

Bettina Selig
Centre for Image Analysis,
Uppsala

Doctoral Thesis
Swedish University of Agricultural Sciences
Uppsala 2015

Acta Universitatis agriculturae Sueciae
2015:16

ISSN, 1652-6880
ISBN (print version) 978-91-576-8230-7
ISBN (electronic version) 978-91-576-8231-4

Print: SLU Service/Repro, Uppsala 2015

Image Segmentation using Snakes and Stochastic Watershed

Abstract

The purpose of computerized image analysis is to extract meaningful information from digital images. To be able to find interesting regions or objects in the image, first, the image needs to be segmented. This thesis concentrates on two concepts that are used for image segmentation: the snake and the stochastic watershed.

First, we focus on snakes, which are described by contours moving around on the image to find boundaries of objects. Snakes usually fail when concentric contours with similar appearance are supposed to be found successively, because it is impossible for the snake to push off one boundary and settle at the next. This thesis proposes the two-stage snake to overcome this problem. The two-stage snake introduces an intermediate snake that moves away from the influence region of the first boundary, to be able to be attracted by the second boundary. The two-stage snake approach is illustrated on fluorescence microscopy images of compression wood cross-sections for which previously no automated method existed.

Further, we discuss and evolve the idea of stochastic watershed, originally a Monte Carlo approach to determine the most salient contours in the image. This approach has room for improvement concerning runtime and suppression of falsely enhanced boundaries. In this thesis, we propose the exact evaluation of the stochastic watershed (ESW) and the robust stochastic watershed (RSW), which address these two issues separately. With the ESW, we can determine the result without any Monte Carlo simulations, but instead using graph theory. Our algorithm is two orders of magnitude faster than the original approach. The RSW uses noise to disrupt weak boundaries that are consistently found in larger areas. It therefore improves the results for problems where objects differ in size. To benefit from the advantages of both new methods, we merged them in the fast robust stochastic watershed (FRSW). This FRSW uses a few realizations of the ESW, adding noise as in the RSW. Finally, we illustrate the RSW and the FRSW to segment in vivo confocal microscopy images of corneal endothelium. Our methods outperform the automatic segmentation algorithm in the commercial software NAVIS.

Keywords: image segmentation, snakes, active contours, stochastic watershed, minimal spanning tree, corneal endothelium, compression wood

Author's address: Bettina Selig, SLU, Centre for image analysis,
Box 337, SE-751 05 Uppsala, Sweden
E-mail: bettina@cb.uu.se

If you can dream it,
you can do it.

—*Walt Disney*

Enclosed Publications

This thesis is based on the following papers, which are referred to in the text by their Roman numerals.

I B. Selig, C. L. Luengo Hendriks, S. Bardage and G. Borgefors, "Segmentation of highly lignified zones in wood fiber cross-sections", in *Image Analysis*, ser. Lecture Notes in Computer Science, A.-B. Salberg, J.Y. Hardeberg and R. Jenssen, Eds., vol. 5575, Springer Berlin Heidelberg, pp. 369–378, 2009.

Selig developed the methods, designed and performed the experiments, and wrote the paper.

II B. Selig, C. L. Luengo Hendriks, S. Bardage, G. Daniel and G. Borgefors, "Automatic measurement of compression wood cell attributes in fluorescence microscopy images", *Journal of Microscopy*, vol. 246, no. 3, pp. 298–308, 2012.

Selig developed the methods, designed and performed the experiments, and wrote the paper.

III B. Selig and C. L. Luengo Hendriks, "Stochastic watershed—an analysis", in *Proceedings of SSBA 2012*, Swedish Society for Automated Image Analysis, pp. 82–85, 2012.

Selig designed and performed the experiments, and wrote the paper.

IV F. Malmberg, B. Selig and C. Luengo Hendriks, "Exact evaluation of stochastic watersheds: from trees to general graphs", in *Discrete Geometry for Computer Imagery*, ser. Lecture Notes in Computer Science, E. Barcucci, A. Frosini and S. Rinaldi, Eds., vol. 8668, Springer International Publishing, pp. 309–319, 2014.

Selig wrote a great part of the program, performed the experiments and co-authored the paper.

V K. B. Bernander, K. Gustavsson, B. Selig, I.-M. Sintorn and C. L. Luengo Hendriks, "Improving the stochastic watershed", *Pattern Recognition Letters*, vol. 34, no. 9, pp. 993–1000, 2013.

Selig developed the methods, designed the experiments and co-authored the paper.

VI B. Selig, F. Malmberg and C. L. Luengo Hendriks, "The fast evalution of the robust stochstic watershed", submitted for publication in a conference proceeding, 2015.

Selig developed the methods, designed and performed the experiments, and wrote the paper.

VII B. Selig, K. A. Vermeer, B. Rieger, T. Hillenaar and C. L. Luengo Hendriks, "Fully automatic evaluation of the corneal endothelium from in vivo confocal microscopy", submitted for journal publication, 2015.

Selig developed the methods, designed and performed the experiments, and wrote the first draft of the paper.

Reprints were made with permission from the publishers.

Related work

While working on this thesis, the author also contributed to the following papers

viii B. Selig, C. L. Luengo Hendriks and G. Borgefors, "Measuring distribution of lignin in wood fibre cross-sections", in *Proceedings of SSBA 2009*, Swedish Society for Automated Image Analysis, pp. 5–8, 2009.

ix B. Selig, M. K. Khan and I. Nyström, "Using a ring-shaped region around the optic disc in retinal image registration of glaucoma patients", in *Proceedings of SSBA 2010*, Swedish Society for Automated Image Analysis, pp. 35–38, 2010.

Contents

1 Introduction

1.1 Image segmentation

The key task in image analysis is to extract meaningful information from a digital image. The area of application is diverse. Image analysis can be used for example to classify soil in remote sensing images [26], to recognize faces [50] using security systems or to detect tumors in medical images [4] to diagnose and plan possible treatment.

To extract information from an image, interesting regions or objects need to be found. For this, the image needs to be partitioned in coherent regions. This procedure is called *segmentation*, and the result is a segmented image, in which each pixel is assigned a label representing an object or the background. Finally, we can obtain information for the labeled regions in form of measurements.

Often correct segmentation is the most difficult task for image analysis applications. A prerequisite for an automatic segmentation algorithm is that it is possible to distinguish between different objects. This is provided when the objects differ, for example, in color, size or texture. Since the input data for the different applications differs strongly from each other, a universal segmentation method cannot be provided. Instead, we have to tailor a suitable procedure to every new problem. For this, a multitude of segmentation algorithms exist which all serve different purposes and are useful for different problems.

If the objects hold different colors, that is intensities in grey value images, various methods, as thresholding [36] or k-means [23], can be applied. If the extent is known to which pixels belonging to the same region differ from each other, region growing methods like statistical region merging [31] are useful.

Often it is beneficial to regard the *gradient magnitude*, which is the difference in grey value of neighboring pixels: where the gradient magnitude is low, neighboring pixels have similar grey values and probably belong to the same object, and where the gradient magnitude is high, the grey value of neighboring pixels differs strongly. In the latter case, an edge in the image is present, which potentially belongs to the contour of an object.

There are many different ways to use the gradient to find the desired outline, for example Canny's edge detector [9] or graph-based methods like Markov Random Fields [43].

In this thesis, we focus on the two segmentation approaches *snakes* [24] and *stochastic watersheds* [1].

Snakes are useful to delineate an object when the appearance and the ap-

proximate location of its contour are known, for example organs in medical images [20]. Snakes are active contour models, which attempt to minimize an energy functional that is based on attributes of the image and the expected shape of the object's contour. For this, the snake is defined as a *spline*, a piecewise polynomial function described by a set of points. It is moved over the image to settle at the position of an energy minimum, which is ideally the object's contour.

The stochastic watershed approach is suitable for the segmentation of structures composed of similarly sized regions and where an approximate number of objects is known, for example multispectral satellite images [33]. The algorithm determines the most salient boundaries in the image using *Monte Carlo simulations*, which means repeated random sampling. Salient boundaries are expected to occur more frequently during the attempts, which are used to determine a final segmentation.

1.2 Thesis outline

Section 2 describes the basic approach of active contour models and proposes the two-stage snake, a variant to find similar boundaries successively.

Section 3 explains the stochastic watershed approach and proposes several versions of the method to overcome two problems, such as high computational costs and falsely enhanced boundaries. Finally, the new methods are compared to each other and to the original approach.

Section 4 shows an application with fluorescence microscopy images of compression wood cross-sections. The appearance of the fibers consists of concentric regions. To segment these regions, we apply the two-stage snake. Afterwards, we compare the segmentation to the delineation done by two experts.

Section 5 concentrates on the segmentation and measurement of in vivo confocal images of human corneal endothelium. We apply two of our proposed stochastic watershed approaches and show that we achieve more accurate results than the commercial software NAVIS.

Finally, Section 6 draws conclusions and discusses possible future work.

2 Segmentation using snakes

2.1 Basic approach

Active contours are methods to delineate the boundaries of objects in an image. These approaches are especially useful when parts of the desired boundaries are missing or otherwise difficult to detect.

The two main active contour methods are level sets [35] and snakes [24]. The level set approach is based on the level set function, which is modified by applied forces. The zero level set, the set of points where the function crosses the xy-plane, is the outline of the desired object(s). This approach has the feature that contours can split and merge during the process. It is often used to track moving objects [37].

A snake is implemented as the spline $v(s) = [x(s), y(s)]$, where $x(s)$ and $y(s)$ are the xy-coordinates along the contour $s \in [0, 1]$. Often the snake is described as a closed contour, since it ought to find the outline of an object. This means $v(0) = v(1)$, the position of the start and the end point of the snake are the same.

The snake approach is suitable if the appearance and the approximate location of the desired contour are known. Snakes are less complex and easier to implement than level sets. Since in the application presented in Papers I and II it is not necessary to split or merge contours, we were able to use the simpler snake approach.

After initializing the snake as close as possible to the expected location of the targeted contour, the snake is moved around to gradually minimize the energy functional E_{snake}, defined as

$$E_{\text{snake}} = E_{\text{int}} + E_{\text{ext}}, \tag{1}$$

where E_{int} is an internal energy that defines the favored shape of the snake, for example short and straight, and E_{ext} is an external energy that is based on the attributes of the image and enables the snake to find the outline of the desired object.

The internal energy E_{int} is defined as

$$E_{\text{int}} = \alpha \left| \frac{\mathrm{d}v}{\mathrm{d}s} \right|^2 + \beta \left| \frac{\mathrm{d}^2 v}{\mathrm{d}s^2} \right|^2, \tag{2}$$

where the parameters α and β regulate elasticity and stiffness, respectively. When increasing α, the first term gains relevance. This means that the snake prefers to be short, acts more like a rubber band around an object and does not necessary delineate concave areas well. When increasing β, the second

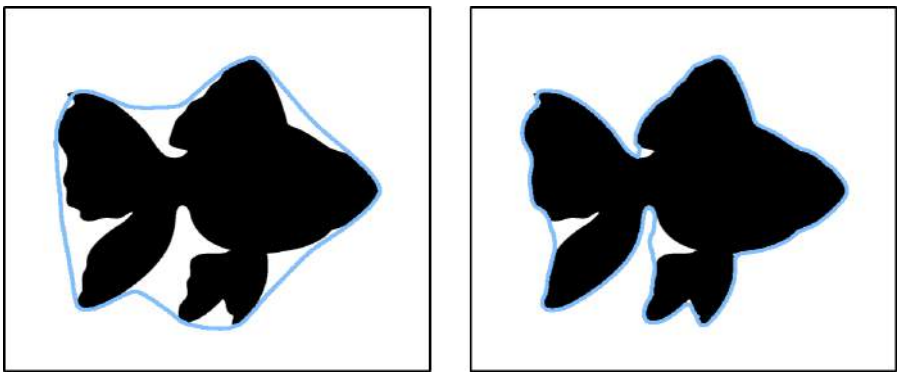

Figure 1: Example for snakes when $\alpha = 1$ and $\beta = 0$ (left), and when $\alpha = 0$ and $\beta = 1$ (right).

term gains relevance, which makes the snake smooth and avoids the formation of sharp edges, see Fig. 1.

The external energy is traditionally based on the gradient magnitude of the image $I(x,y)$, as

$$E_{\text{ext}} = -|\nabla I(x,y)|^2. \tag{3}$$

This makes the snake settle in regions with high gradient, that is boundaries.

The forces that are applied to the snake during the minimization process are the derivatives of the present energies:

$$\vec{F}_{\text{int}} = -\frac{\nabla E_{\text{int}}}{||\nabla E_{\text{int}}||} \tag{4}$$

and

$$\vec{F}_{\text{ext}} = -\frac{\nabla E_{\text{ext}}}{||\nabla E_{\text{ext}}||}. \tag{5}$$

The gradient vector flow (GVF) [49] produces an alternative external force that is more suitable if the snake is initialized far away of the desired contour or the contour has concave sections. A GVF field is created that applies a constant (normalized) force on each point of the image. To determine the GVF field, we need to find the force $\vec{F}_{\text{GVF}} = (U_{\text{GVF}}, V_{\text{GVF}})^T$ that minimizes the energy functional

$$E_{\text{GVF}} = \int \mu(|\nabla U_{\text{GVF}}|^2 + |\nabla V_{\text{GVF}}|^2) + |\nabla E_{\text{ext}}|^2 |\vec{F}_{\text{GVF}} + \nabla E_{\text{ext}}| \mathrm{d}x, \tag{6}$$

where μ is to balance the influence of the green and blue terms. The blue term in the energy functional E_{GVF} ensures that the direction of the force is

equal to the direction of the earlier defined external force $\vec{F}_{\text{ext}}$ in regions of high gradient, which means close to edges. In regions of low gradient, $\vec{F}_{\text{ext}}$ contains too little information and is not able to push the snake towards desired contours. The green term in the function for E_{GVF} makes these regions smooth, which means that the information of the boundaries of the regions is spread over the whole region. The result is that boundaries have a larger influence region and the snake will be attracted to them even when it is initialized far away. Additionally, the snake improves delineating the contour of an object, if it has a concave outline.

If only the internal force is applied to the snake and the snake is initialized inside the object, the contour of the model shrinks and finally collapses. To make sure that the snake is pushed outwards to reach the desired boundary, an additional force, called balloon force, is often used. A small force is applied orthogonal to the outline of the snake. It is added to $\vec{F}_{\text{ext}}$, alternatively to $\vec{F}_{\text{GVF}}$, as

$$\vec{F}_{\text{ext+balloon}} = \kappa \vec{F}_{\text{ext}} + \kappa_{\text{p}} \vec{n}(s). \tag{7}$$

The parameter κ is the weighting of the external force and κ_{p} is the weighting of the balloon force.

Since the snake approach builds on energy minimization using gradient descent, there might not be an optimal final position for the snake and after a while the snake might oscillate between two locations. One can define an ϵ that determines a minimal energy left in the system or a maximal distance moved within the last steps. But this parameter is not always easy to define and when the snake overshoots the desired boundary and does not settle at all, it is anyway necessary to have a maximal number of iterations. Therefore, sometimes it is easier to have only a maximal number of iterations as stopping criterion.

2.2 Two-stage snakes

When using snakes to successively find boundaries of concentric regions, we face a problem. If the desired boundaries have similar conditions, for example, are transitions from darker to lighter regions, it is impossible for the snake to push off the first boundary and then settle at the second. This is because the same forces that make the snake move away from the first boundary, prohibits stopping the ongoing movement at the second boundary and instead pushes the snake further away.

To overcome this problem, we introduced an intermediate step in Paper I: after the snake traced the outline of the first boundary, we apply a

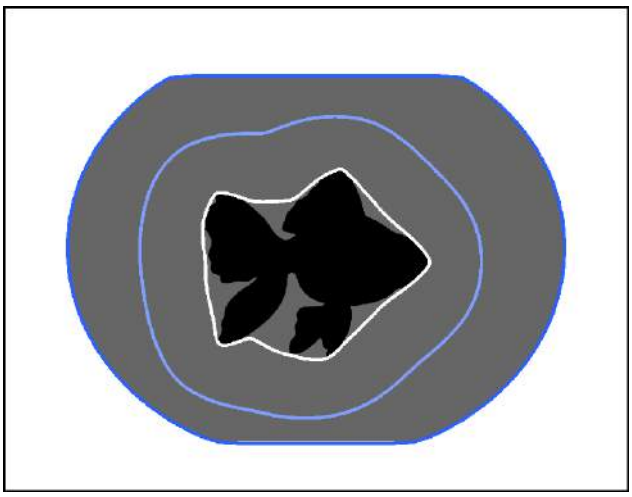

Figure 2: Illustration of two-stage snake: Initialization (white line), intermediate steps (light blue line) and final segmentation (dark blue line).

new external force to the snake (with an additional balloon force). We call this new force *complementary force*. The corresponding energy describes a minimum between the two desired boundaries, as for example $-F_{\text{ext}}$.

The result of applying the complementary force is that the snake moves outwards (due to the balloon force) to settle at a position between the two desired boundaries. From this new position, the initial external force can be applied again, so the snake is attracted by the second, the outer boundary, see Fig. 2. This approach has the advantage that no additional parameters are needed. Only the complementary force needs to be determined.

A prerequisite for this new force is that the complementary energy has its minimum between the desired contours. Otherwise the snake passes the contour during the intermediate step and then cannot retract in the final step, due to the added balloon force.

3 Segmentation using stochastic watershed

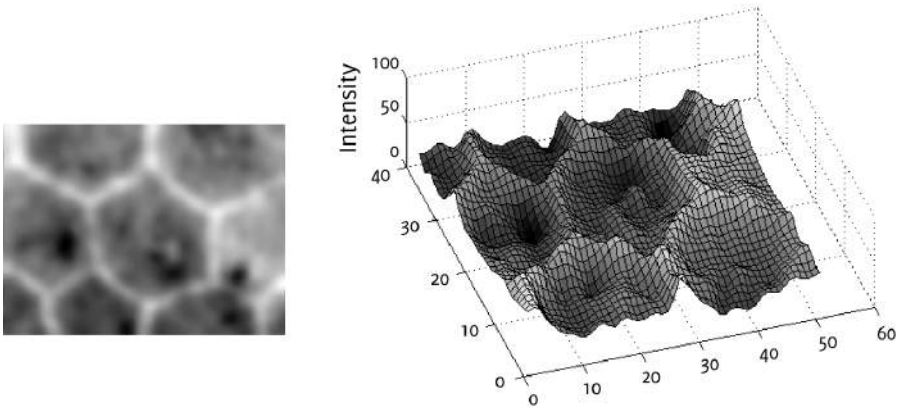

Figure 3: Endothelial cell image (left) and representation as geodesic landscape (right).

3.1 Basic watershed methods

In the *watershed segmentation* algorithms [6], we regard the image as a geodesic landscape. Low gray level values correspond to valleys and high values to hills, see Fig. 3. When the water level raises, each local minimum functions as water source, since water emerges from it. The landscape is subsequently flooded. Eventually, the water of two minima meet at the ridgeline between their catchment basins, where now a watershed line is placed. Hence, the watershed segmentation separates regions that yield precisely one minimum.

Digital images, as e.g. microscopy images, usually suffer from noise, so that the desired region often contains more than one local minimum. Therefore, the basic watershed approach often yields an over-segmented result, see Fig. 4.

To reduce the number of segmented regions, we can use the *H-minima transform*, see Fig. 4. Here, all minima that have a depth less than h are suppressed and therefore are not used as sources during the immersion process. The idea to remove the most shallow minima is based on the assumption that these minima are least important. But due to non-uniform lighting conditions or different amounts of contrast in the image, this method alone is sometimes not sufficient to give satisfactory segmentation results.

In case the rough positions of the desired regions are known, *seeded watershed* can be used, see Fig. 4. First, we set a seed point for each region

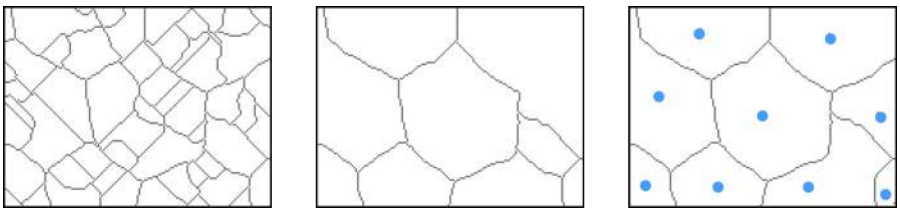

Figure 4: Segmentation results of example image in Fig. 3 using different watershed approaches: watershed segmentation (left), watershed segmentation using H-minima (middle), seeded watershed segmentation (right). Dots show the locations of the seed points.

manually or automatically. These seed points then function as sources for the flooding operation described above, whereas the local minima are ignored. However, positioning the seed points is not a trivial task. Using a manual approach is not feasible for many applications, because it is too time consuming. And finding the regions automatically transforms the segmentation task into a problem of object detection.

Later, we concentrate on more elaborate watershed methods, but first, we explain how a basic watershed segmentation can be obtained.

3.2 Concepts to determine a watershed segmentation

There are several algorithms to obtain a watershed segmentation of a given image. In this section, we first regard two attempts that simulate the idea of flooding a landscape, and later discuss the possibility of transferring the problem to graph theory.

The concept of raising the water level of the image (immersion simulation), is mimicked by an algorithm published by Vincent and Soille [46]. Here, we describe a variation of this method.

First, we sort all pixels in the image according to their gray level. Then, we regard the pixel v with the smallest value and remove it from the sorted list. To determine if v belongs to one region or to a watershed line, we regard the following three cases: 1) If none of v's neighbors is labeled, create a new label for v. 2) If there exists only one label in the neighborhood, we label v with that label. 3) If the neighborhood contains more labels, we mark v as watershed pixel. We proceed with the algorithm by continuing processing the pixels in the sorted list, until the list is empty and all pixels are either labeled or marked as watershed pixels.

Another more common approach is the flooding simulation introduced by Beucher and Meyer [7]. It is based on the idea that the water emits from

water sources, which are either all local minima in the image (standard watershed) or a set of chosen seed points (seeded watershed). For simplifications, we call these water sources *markers*.

As a first step, we label all markers individually. Next, we add all neighboring pixels of the initial markers in a priority queue, where the priority is the gray level value of the pixel. Now we visit and extract the pixel v with the highest priority (lowest gray value) from the queue. If all labeled neighbors of v have the same label, v is also labeled with their label. If the labeled neighbors have different labels, v is marked as a watershed pixel. Pixels of the neighborhood that have not been visited yet are now inserted in the priority queue. We continue to process the pixels in the priority queue until the queue is empty and all pixels in the image are visited.

A digital image can be transferred to a edge-weighted graph G, where vertices V correspond to the pixels in the image and the weights of the edges E to the similarity of the pixels, e.g. the difference in grey values. In this representation, the watershed segmentation can be solved by *watershed cuts*, introduced by Cousty et. al [13].

Where the watershed in an image is defined by pixels that form the boundaries between the regions, a watershed cut is a set S_G of edges that, when removed, splits the graph G into two or more disjoint connected components (representing the found regions).

To simplify the problem, we only regard the *minimal spanning tree* (MST) T. The MST is a subgraph of G that connects all vertices V and in addition, has a minimal total edge weight.

The result of the watershed cut algorithm is a *minimal spanning forest* (MSF) G_F of the original graph, where each connected component contains exactly one marker. This MSF is not only a subgraph of G, but also a subgraph of T. Since the regions of the watershed segmentation are defined by markers, the watershed cut S_G of the complete graph corresponds to watershed cut S_T of the MST. The MSF G_F can be computed by any algorithm creating an MST T with some additional conditions [28].

There are two basic algorithms that are often used to create the MST of a graph. They were introduced by Kruskal [25] and Prim [39], respectively. When inspecting the procedure of these two algorithms, one finds correlations to the two concepts of computing a watershed segmentation, described above. Kruskal's approach functions similar to the immersion principle and Prim's algorithm to the flooding simulation method.

When producing the MSF instead of the MST, we need to alter the original algorithms slightly: to ensure that each connected component contains exactly one marker, we do not include edges of the MST that establish

a path between markers in the resulting graph.

When creating an MSF using an algorithm based on Kruskal's idea, we proceed as follows: first, we set all vertices as unlabeled and sort all edges according to their weights. Next, we regard the edge e_{vw} with the smallest weight (connecting v and w) and remove it from the sorted list. To determine if e_{vw} belongs to the MSF, we consider the following three cases: 1) If both vertices v and w are not labeled, create one new label for both vertices and include e_{vw} to the MSF. 2) If only one vertex is labeled, label the other vertex with the same label and include e_{vw} to the MSF. 3) If both vertices v and w are labeled, do not include e_{vw} in the MSF. Including this edge would either form a cycle in the graph or connect components with different local minima. Afterwards, continue proceeding with the next edge from the sorted list until all edges have been processed and the list is empty.

Contrary to Kruskal's method, with Prim's algorithm we can set markers from where the regions expand. To create an MSF with a Prim-like approach, we first label all markers with an individual label. Next, we add all edges adjacent to the markers in a priority queue, where the priority is the weight of the edges. Then, we visit and extract the edge e_{vw} with the highest priority (lowest weight) from the queue. If only one vertex, v or w, is labeled, we label the other vertex with the same label and include e_{vw} to the MSF. We also enqueue the edges connected to the newly labeled vertex in the priority queue. But if both v and w are already labeled, we do not include e_{vw} in the MSF. Including this edge would either form a cycle to the graph or connect components that contain a marker each. Now, we proceed with the next edge from the priority queue until all edges have been visited and the queue is empty.

The labeled components in both methods now represent the segmented regions. To create a watershed line between the regions in the image representation, we can, for example, mark pixels that are neighbors to pixels with a different label as watershed pixels. As a result, the watershed lines are two pixels thick.

3.3 Stochastic watershed

The ideal segmentation algorithm produces a perfect segmentation result without the user providing any previous knowledge. Unfortunately, this is not a realistic vision, since it has to be defined how the important object(s) can be distinguished from the rest of the image.

More realistic is to develop an algorithm with a small number of parameters, which also are intuitive to select. The *stochastic watershed*, which was proposed by Angulo and Jeulin [1], is moving in this direction.

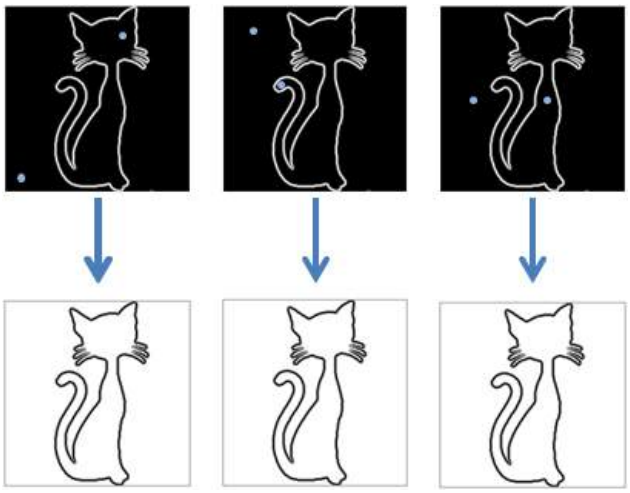

Figure 5: Gradient magnitude with seed points placed in different regions (top) and segmentation results (bottom).

This algorithm has three parameters: the number of seed points N, the number of realizations M and the number of most significant regions R. The number of seed points N and the number of most significant regions R are directly related to the number of objects that ought to be segmented. The number of realizations M defines how precise the segmentation result is in relation to the time spent on the calculations.

The stochastic watershed utilizes a property of the seeded watershed, which is that the algorithm is very insensitive to the placement of seed points within a region: Considering two neighboring regions, their seed points can move large distances in each respective region and the result still yields the same segmentation line. This is due to the order in which the pixels are processed, see Fig. 5.

The stochastic watershed uses Monte Carlo simulations to find salient boundaries. Here, seeded watershed with randomly placed seed points is applied repeatedly to the image. Whether the boundary between two neighboring regions will be detected depends on the random placement of the seed points in these regions. There are two cases: 1) Each of the regions contain (at least) one seed point, or 2) only one or none of the regions contain seed points.

In the first case, the boundary between the regions is found, due to the property of the seeded watershed, described above. In the second case, the boundary between the two regions will normally not be found at all or only found as fragments, see Fig. 6. (Sometimes, when seed points in other parts

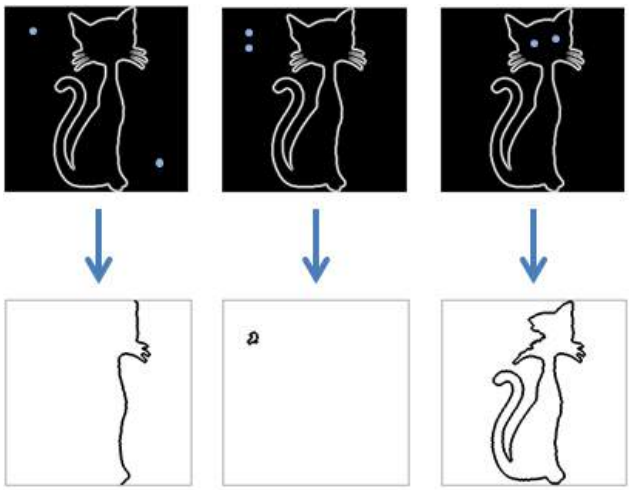

Figure 6: Gradient magnitude with seed points placed in the same region (top) and segmentation results (bottom).

of the image are placed advantageously, the boundary can be found anyway, but for now, we refrain from these cases.)

The probability is high that salient boundaries are found when applying seeded watershed with randomly placed seed points. When this step is proceeded repeatedly, a probability density function (PDF) of the boundaries can be determined.

In order to obtain a segmentation using stochastic watershed, Angulo and Jeulin [1] create M realizations of seeded watershed with N randomly placed seed points. They then construct the PDF combining the M segmentations using the Parzen window method. In our versions of the algorithm, we simply add the segmentation results to create the PDF.

Finally, Angulo and Jeulin segment the PDF with volumic watershed for defining the R most significant regions. This last step of segmenting the PDF into R regions, we changed in our versions to a standard watershed segmentation with H-minima transform, where all minima are erased that are smaller than a threshold h. This seems to be more fitting to many applications, because we do not persist in having exactly R regions in the final segmentation. Especially when the image contains a multitude of objects, the user is often only able to provide a rough estimate for the number of desired objects. Even though the parameter h in the H-minima transform is barely intuitive to choose, this approach often yields a better segmentation result. As described before, the H-minima assumes that the most shallow minima are the least important. Since we are operating on an image repres-

enting the significance of boundaries in the image, this assumption is true and therefore the H-minima transform is suitable to apply.

In the remainder of this thesis, when we refer to the stochastic watershed segmentation algorithm, we mean a version of the original method where we determine the PDF by simply adding the M seeded watershed segmentations and create the final result using the standard watershed with H-minima transform.

3.4 Parameters

In Paper III, we focused on the first part of the stochastic watershed, where the PDF is created, and studied the relation between the algorithm's parameter N and the actual number of regions in the image. We discovered that the value of N has a great influence of how many realizations of the seeded watershed are needed to make a good segmentation of the PDF possible.

To show this, we created three synthetic images with different numbers of equally sized regions. With these images and different values for N, we determined the minimal number of realizations $M_{\min}$ to create a PDF, where the original boundaries can be distinguished from the background through thresholding. From the results, we concluded that the algorithm is not very sensitive to the value of N, but that when choosing N close to the actual number of regions in the image, $M_{\min}$ is lowest. This means that the distinction between boundaries and background is possible with fewer realizations.

The disparity of boundary and background values in the PDF grows over time (according to our experiments). Hence, the smaller $M_{\min}$ is, the easier it is to achieve a good segmentation of the final PDF (after M realizations).

Even though we noted that the stochastic watershed is not very sensitive to the value of N, it cannot be set arbitrarily, but needs to be chosen for the application by visual inspection or training. If N is set too small, small details in the image can be lost, and by choosing it too large, weak boundaries could be found repeatedly, which results in an over-segmentation.

The parameters M and h are dependent on how difficult it is to segment the desired objects in the image. If it is simple to detect the boundaries, a suitable segmentation can be determined after only a few realizations of the seeded watershed. Additionally, the algorithm is not that sensitive to the threshold h, due to the disparity of boundary and background values in the PDF mentioned before. However, if the boundaries are unclear, we need more realizations to determine a reliable result. Because of this, the value for M is often set to 50 or 100 [16, 34]. Additionally, the algorithm

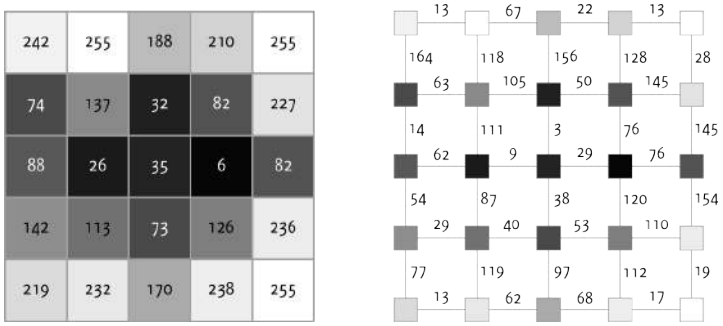

Figure 7: Original image (left) and edge-weighted graph G (right). The value of the edge weights is the absolute difference of the gray value of the neighboring pixels connected by the edge.

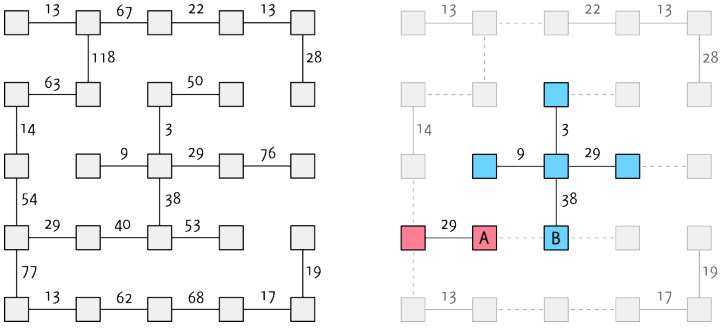

Figure 8: MST T (left) and forest G' (right). To construct G' all edges with weights greater or equal to the weight of e_{AB} are removed from T. The trees T_A and T_B are presented in red and blue, respectively.

gets more sensitive to h, because real boundaries and falsely enhanced lines might not be separable in the PDF.

3.5 Exact evaluation of the stochastic watershed

Due to the multiple realizations of seeded watershed that are needed to obtain a useful estimate of the PDF, the SW is not very fast. For this, Meyer and Stawiaski [29] worked on a way to determine the PDF exactly, without Monte Carlo simulations. Malmberg and Luengo Hendriks [27] extended this approach and presented an efficient algorithm to calculate the exact PDF.

Meyer and Stawiaski [29] developed a formula to calculate the probability that neighboring pixels are in different regions, when using randomly

placed seed points. For this, we regard the MST T of the edge-weighted graph G representation of the input image, see Fig. 7.

We explain the concept by determining the probability that the vertices A and B in Fig. 8 are in different regions. That means that edge e_{AB} connecting A and B is included in a watershed cut S_T. In our example, the edge e_{AB} has the value 40. Since all edges greater or equal to this value are not relevant for this case, we remove all these edges from T. The result is a forest G', in which one of the trees contains vertex A and another one contains vertex B. These trees we call T_A and T_B, respectively, see Fig. 8.

The probability that A and B lie in different regions corresponds to the probability that seed points fall in both the area presented by T_A and T_B. To compute this probability, we consider that $\rho(v)$ is the probability that the pixel represented by vertex v is a seed point. If all pixels have the same probability to be chosen as a seed point, then $\rho(v) = \frac{1}{|V|}$, where V is the set of all vertices.

Hence, the probability that a seed point is placed within a subtree T' is

$$\widetilde{T'} = \sum_{v \in V(T')} \rho(v). \tag{8}$$

This means that the probability that A and B lie in different regions, when using N randomly placed seed points, is

$$F(\widetilde{T_A}, \widetilde{T_B}) = 1 - (1 - \widetilde{T_A})^N - (1 - \widetilde{T_B})^N + (1 - \widetilde{T_A} - \widetilde{T_B})^N, \tag{9}$$

where the blue and red terms specify the probability that there is no seed point in T_A or T_B, respectively, and the green term describes the probability that in neither of the trees T_A or T_B there is a seed point. The green term needs to be added as an adjustment, since we considered the case that there is no seed point in both trees, T_A and T_B, twice (in the blue and the red term).

Malmberg and Luengo Hendriks [27] developed an efficient, quasi-linear algorithm to enable this approach. The efficiency of this method lies in the usage of disjoint-set data structures. These data structures keep track of connected components formed when edges are added to a graph.

The proposed algorithm works as follows: after transforming the image into a edge-weighted graph G and determining the MST T, all edges of T are sorted and stored in a list C. First, the edge e_{AB} with the minimal weight (connecting the vertices A and B) is regarded. To set a probability value for this edge to be included in a watershed cut S_T, we consider the subgraph G', an MSF that is the MST excluding the edges in C. More precisely, we

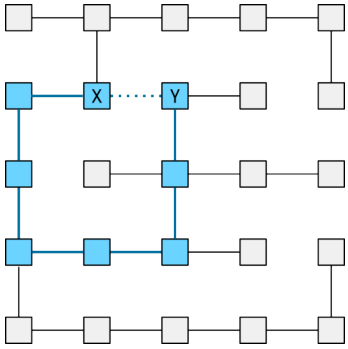

Figure 9: Tree T_{PDF} (weights not shown).

regard the connected components (trees) T_A and T_B. Now, we can calculate the probability $F(\widetilde{T_A}, \widetilde{T_B})$ that e_{AB} is in a watershed cut S_T. Finally, we extract e_{AB} from C and continue with the next smallest edge in C until C is empty and all edges are processed.

The result of this algorithm is a tree T_{PDF} with probability values as edge weights. This representation of the PDF is difficult to visualize. For further processing, it can be useful to transform the tree to a PDF image. In Paper IV, we propose a method to extend the tree representation of the PDF to a graph G_{PDF} that can be further processed to an image.

We illustrate our approach on the tree T_{PDF} shown in Fig. 9. For this, we consider the edge e_{XY}, which connects the vertices X and Y in G_{PDF}, but is not present in T_{PDF}. Further, we regard the path π_{XY} (blue in Fig. 9) connecting X and Y in T_{PDF}.

If X and Y are in different regions, at least one edge along π_{XY} must have been cut. Therefore, the probability that e_{XY} is included in a watershed cut $S_{G_{\text{PDF}}}$, corresponds to the probability that at least one edge along π_{XY} is included in the watershed cut $S_{T_{\text{PDF}}}$. Therefore, the probability is defined as

$$P(e_{XY}) = 1 - \prod_{e \in E(\pi_{XY})} (1 - P(e)). \tag{10}$$

In Paper IV, we propose an efficient algorithm that calculates this probability for all edges in the graph G in linear time. For this, we first define a function

$$\phi(X, Y) = 1 - P(e_{XY}) = \prod_{e \in E(\pi_{XY})} (1 - P(e)). \tag{11}$$

Additionally, we choose an arbitrary vertex in T_{PDF} to be the root R. We define $\phi_R(v) = \phi(v, R)$ for any vertex v in the tree.

We now can calculate the probability of e_{XY} being included in a watershed cut S_G by

$$\phi(X,Y) = \frac{\phi_R(X)\phi_R(Y)}{(\phi_R(\mathrm{LCA}(X,Y)))^2}, \tag{12}$$

where $\mathrm{LCA}(X,Y)$ is the *lowest common ancestor* of the two vertices X and Y. An ancestor of a vertex v is a vertex that lies on the direct path to R. Further, a lowest common ancestor of vertices v and w is ancestor to v as well as to w and lies farthest away from R.

In our algorithm, we first calculate $\phi_R(v)$ by traversing all vertices v via breadth-first search. Afterwards, we calculate $P(e_{vw})$ for all edges in G_{PDF}. For this, we need to determine $\mathrm{LCA}(v,w)$. Using the algorithm by Bender and Farach-Colton [5], this step can be done in constant time, after an $\mathcal{O}(|V|)$ preprocessing step. This concept is based on an idea by Harel and Tarjan [21] and is mainly responsible for the efficiency of the ESW.

To finally transform G_{PDF} to an image, we need to determine for each pixel v how probable it is that a pixel w in the neighborhood $\delta(v)$ lies in a different region. This corresponds to the probability that at least one of the edges connected to v is included in a watershed cut S_G, which is defined as

$$P_G(v) = 1 - \prod_{w \in \delta(v)} (1 - P(e_{vw})). \tag{13}$$

3.6 Robust stochastic watershed

In Paper III, we first observed that the stochastic watershed works best when the regions in the image are similarly sized. Since the probability of placing a seed point is the same over the image, large areas often receive several seed points and small regions possibly none. This means that the algorithm tends to split large area and miss small ones.

To understand the origin of falsely enhanced lines, we regard the case that one region contains two seed points. In Fig. 10(b), the seed points are placed as far from each other as possible. To find the boundary between the two regions specified by the seed points, neighboring pixels to the emerging regions will be successively assigned to one of the seed points. The order in which the pixels are assigned is defined by their gray values, as described before. This specific order does not change when we position the seed points elsewhere in their respective regions, see other examples in Fig. 10(c-e). This means that the algorithm will always find the same boundary as long as there is one or several seed points present in each of the regions. This happens regardless of whether there was an actual boundary located in the ori-

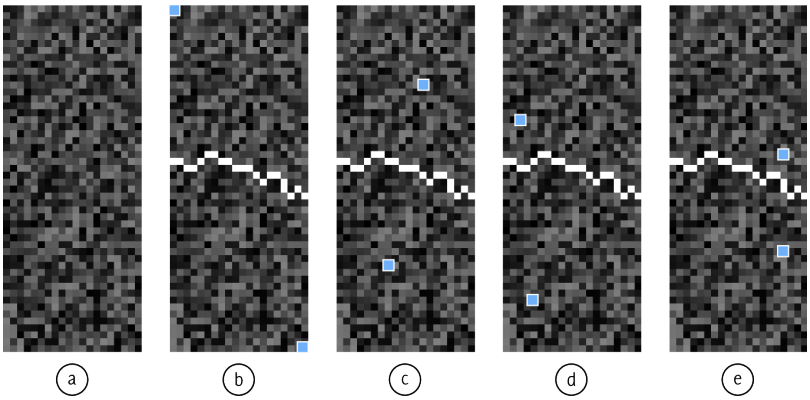

Figure 10: Examples to illustrate that a boundary does not change when changing positions of seed points.

ginal image or if the algorithm falsely enhances a non-existing boundary and thus fabricates a so called *false boundary*.

The insensitivity to the placement of the seed points usually enables the stochastic watershed segmentation to find reliably relevant boundaries, but, in the case of false boundaries, it works to our disadvantage.

To handle this issue, we introduced two improvements presented in Paper V. Firstly, we add noise to the image before each realization of seeded watershed, and secondly, we place the seed points in a grid with random offset and rotation.

The reason for the occurrence of false boundaries is that each realization uses the exact same order in which the pixels are assigned to regions. One way to alter the order in which the pixels are processed, is to add a small amount of noise to the original image before each realization. This does not affect the image boundaries, but mixes up the local rank order. In our experiments, we received best results with noise stronger than the noise in the input image, but not strong enough to hide the signal.

In Paper V, we noted that the amount of noise can be determened with the help of a noise estimation of the image, e.g. using the method by Immerkaer [22]. However, in later experiments (not reported), we did not find any correlation between the optimal amount of noise and the estimated noise strength in the input image.

The idea behind the second improvement, placing the seed points on a grid, is the following: when seed points are placed directly next to each other (as it happens using the Poisson process), a segmentation line is forced to be set between them. The idea of placing them in a grid, is to increase

their distance to each other as much as possible, so the influence of the image values on the segmentation is maximized and the dependency of the seed point positions is minimized.

However, our experiments showed that the grid (square or hexagonal) only improved the segmentation in combination with the first modification. Without adding noise to each realization, the formation of false boundaries dominated the PDF in such a way that the benefit was not apparent.

Just as in the SW, the optimal number of seed points in the RSW lies around the actual number of regions in the image. Using fewer seed points has similar effects on both methods: the number of realizations needed increases. When increasing the number of seed points, the SW breaks down, since at some point false boundaries dominate the PDF. In contrast, the RSW was less affected by the number of seed points and still produced useful PDFs using four times the optimal number of seed points. We concluded that the RSW is less sensitive to the choice of N compared to the SW. By the way, the sensitivity of the RSW to the parameters M and h did not differ from the SW.

3.7 Fast evaluation of the robust stochastic watershed

With the RSW, we overcome a major drawback, namely the tendency of the SW to segment the image into similarly sized regions. But since the RSW also uses the idea of Monte Carlo simulations, the runtime is similar to the SW. With the ESW, we could speed up the SW algorithm, but this method does not seem easily compatible with the RSW.

While the RSW requires several realizations of the seeded watershed, the ESW works on a graph representation, more precisely the MST, of the image. To imitate the idea of adding noise to the original image, like in the RSW, we would need to combine several trees in one graph. This is not a trivial task. In Paper VI, we developed a hybrid approach instead and called it the *fast evaluation of the robust stochastic watershed* (FRSW).

Using the ESW approach, seed point placement is no longer an issue, so we ignore the advantage of placing the seed points in a grid. Further, we abandoned the idea of incorporating the addition of noise in the ESW, and instead perform the ESW several times by using the original image with a small amount of noise added.

The RSW typically uses about 50-100 realizations of seeded watershed. This number of realizations M is chosen to match the conditions of the SW. The ESW determines the result for $M = \infty$ within one step. Since we use the ESW repeatedly in our approach published in Paper VI, the num-

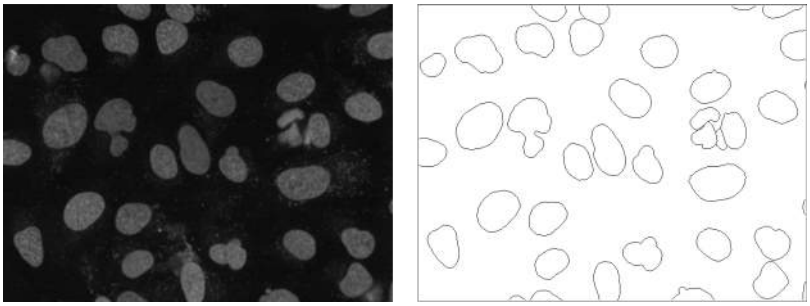

Figure 11: Example image of data set (left) and its hand-drawn ground truth (right).

ber of realizations M_f can be much smaller and still produce good results. The value of M_f is only related to the required number of noise images to suppress false boundaries.

To get the most information out of the M_f intermediate PDFs as possible, we merge them in an advantageous way to obtain the final PDF. In Paper VI, we found the geometric mean a suitable method, since lines (even though they might be weak) present in most PDFs are preserved and lines present in only a minority of the PDFs are suppressed. The geometric mean of the pixel x_i at position i in the final PDF is determined as

$$x_i = \left(\prod_{m=1}^{M_f} x_{i,m} \right)^{\frac{1}{M_f}}, \tag{14}$$

where $x_{i,m}$ is the pixel at position i in the m-th intermediate PDF.

In our experiments, we showed that the FRSW achieves segmentation results on par with the RSW, when using $M_f = 3$ (compared to RSW using $M = 50$). Since the complexity of the calculation of one realization of ESW is similar to one realization of seeded watershed, the reduction of M_f leads to a considerable speedup of the algorithm. In our calculations using single core of an Intel Xeon CPU X5650, the segmentation of each of the images in the data set with size 675×515 pixels took in average 6.4 s seconds for the FRSW, where it took in average 42.8 s for the RSW.

3.8 Comparison of presented stochastic watershed methods

In this section, we apply the four presented stochastic watershed methods SW, ESW, RSW and FRSW to a data set of fluorescence microscopy images of nuclei and compare the results with the hand-drawn ground truth. The

Table 1: Optimal parameters for different methods. Since we used the leave-one-out method, the stated values are the optimal parameters for the great majority of the images in the data set.

Method	Realizations	Seed points N	Noise s	Threshold h
SW	50	50	-	0.15
ESW	-	2	-	0.005
RSW	50	250	0.1	0.2
FRSW	3	100	0.05	0.05

46 images and the ground truth are provided by Coelho [12], see example image in Fig. 11. Note that the task of segmenting the objects in these images is not difficult, one probably can solve it with some mathematical morphology operations and thresholding. The reason to use this data set is simply to illustrate abilities and challenges using the different stochastic watershed approaches. Parts of the result were shown in Paper VI.

First, we needed to find the optimal parameters for the methods. In Table 2, we summarized the parameters needed for each approach. We chose $M = 50$ and $M_f = 3$. The rest of the parameters we determined by training. For this, we ran the algorithms with different sets of parameters, where the parameter N could take eleven different values from 2 to 300, s seven different values from 0.005 to 0.5 and h eleven different values from 0.0005 to 0.4. We compared the segmentation results with the ground truth using the *F-measure*, as in Arbelaez et al. [2]. The F-measure is the harmonic mean of precision and recall and determines the quality of the segmentation results. It can take values between 0 and 1, whereas a values close to 1 means that the segmentation result resembles the ground truth. For this measure we consider segmentation line pixels as a mismatch when they are located four or more pixels away from the corresponding boundary. We define the optimal parameters for each method and for each image as the set of parameters with the highest mean F-measure using the leave-one-out method.

Afterwards, we obtained the segmentation results for each method for the whole data set using the corresponding optimal parameters. In Fig. 12, we show the final PDFs and the segmentations for the example image in Fig. 11.

Finally, we determined the F-measures of the results, see Fig. 13. As expected, the SW and the ESW introduce a lot of false boundaries, which leads to low F-measures, whereas the RSW and the FRSW enhance only the actual boundaries in the image and therefore achieve high F-measures.

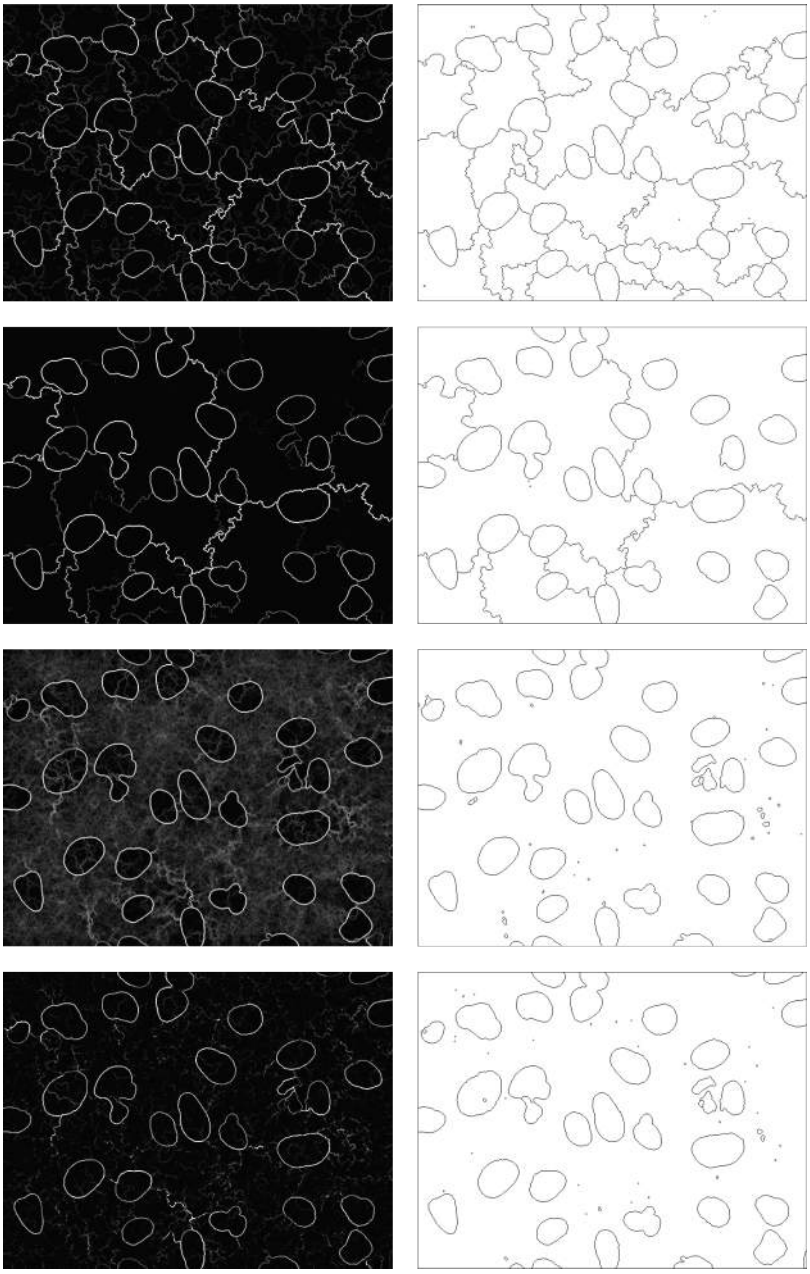

Figure 12: Final PDF (left) and segmentation (right) using SW (top row), ESW (second row), RSW (third row) and FRSW (last row).

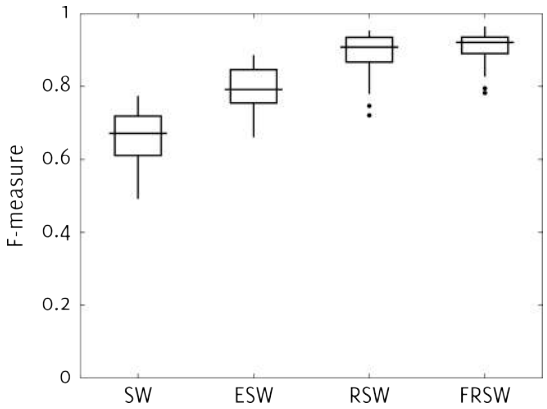

Figure 13: F-measures for segmentation results.

We then examined how many realizations were needed for the different methods to converge to their optimal result. For this, we chose a representative image (Fig. 11) from the data set that achieved average F-measures for all methods. We performed the SW, RSW and FRSW with the optimal values for the parameters N, s and h determined above, and $M = 50$ and $M_f = 50$, respectively. After every one of the 50 realizations, we obtained a segmentation result. The F-measures of these segmentations show how fast each method converges to its best result. The ESW is not built on Monte Carlo simulations and therefore reaches its optimum within one step. In Fig. 14, we show the average F-measures for ten runs of SW, RSW and FRSW, and additionally the F-measure of ESW.

In this experiment, SW stabilizes after 15 realizations, RSW after 25 and FRSW already after two. RSW and FRSW converge to the same value. Even though we expected the same behavior for SW and ESW, the F-measure for SW settles at a lower value. The reason for this lies in the chosen parameters. The optimal number of seed points for ESW is $N = 2$. This means, if we would run SW with $N = 2$ for an infinite number of realizations, it would converge to the same F-measure as ESW. But with the constraint of $M = 50$, SW reaches its maximal F-measure with $N = 50$, which was determined as optimal number of seed points, see Table 1.

In Fig. 14, you can observe that the function for the RSW is serrated. This is due to the value chosen for h and is explained in detail in Paper V.

During each realization, both the SW and the RSW perform a seeded watershed segmentation, whereas the FRSW creates a complete PDF using the ESW method. The ESW is more complicated and time consuming than a simple watershed, so that is not completely obvious that FRSW has a

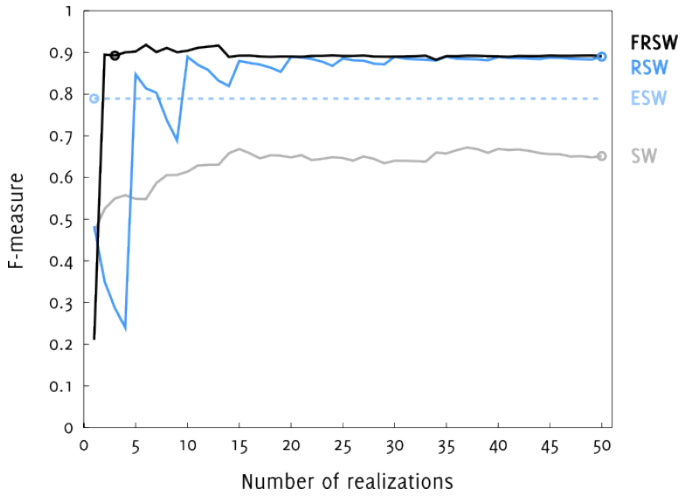

Figure 14: Mean F-measures for ten segmentation attempts for each of the presented methods. The F-measures were determined after each of the 50 realizations.

Table 2

Method	Parameters	F-measure	Runtime in s	Deterministic
SW	N, M, h	0.67	40.1	No
ESW	N, h	0.79	2.3	Yes
RSW	N, M, s, h	0.91	43.5	No
FRSW	N, M_f, s, h	0.92	6.4	No

shorter runtime.

In Table 2, we listed the average runtime for the methods, performing the segmentations of the 46 images of the data set. The SW and RSW need around 40 seconds for a segmentation, whereas the RSW is a bit slower than the SW due to the additional computations made for adding noise and placing the seed points in a grid. The fastest method is the ESW with 2.3 seconds runtime. However, the FRSW is also comparatively fast with 6.4 seconds. This means that the small amount of realizations for FRSW overcomes the fast computation of seeded watershed in RSW.

The ESW differs a lot from the other presented methods, since it is the only one that is not based on Monte Carlo simulations. This makes the runtime constantly fast and also the segmentation result deterministic.

4 Application: Fluorescence microscopy images of compression wood

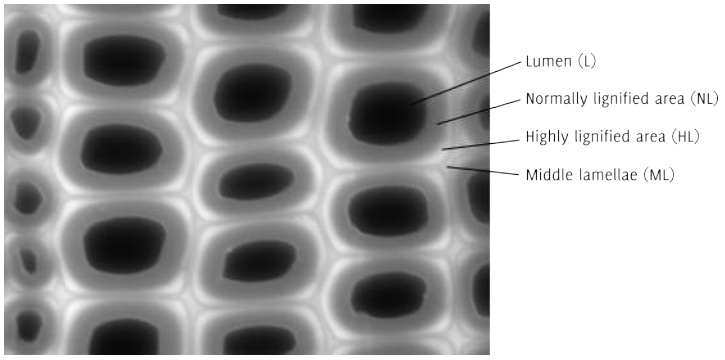

Figure 15: Fluorescence microscope image of compression wood fiber cross-sections.

4.1 Background

In the wood and fiber product industry, the property of the wood fibers is essential for their further processing. The fibers' condition determines if the raw material is suitable, and what it is suitable for. To examine the characteristics of wood (fibers) different imaging methods can be applied and various image analysis approaches have been proposed [3, 30, 32, 48]. The advantage of automatic measurements over manual delineation is that the results are deterministic and fast to obtain for a great amount of images.

In Papers I and II, we concentrated on compression wood, which is an abnormal reaction wood of softwood (i.e. conifers) that is formed through physical stress. It has limited value in the forest product industry, e.g. due to its different mechanical properties [38].

Wood fibers are hollow, up to 2 mm long and have a diameter of approximately 30 μm. The fibers consist of cell wall and lumen (hollow center). The area between the fibers is occupied by the middle lamellae. In normal wood, the middle lamellae contains a high concentration of lignin compared to the cell wall. Lignin is a substance that contributes to the stability of wood. In compression wood, a greater lignification of the cell walls take place, starting from the corners of the cells towards their lumens [14].

Since lignin is autofluorescent, its distribution can be made visible with fluorescence microscopy [15]. In Fig. 15, compression wood fiber cross-sections are shown that were captured by a fluorescence microscope. You

can distinguish between lumens, areas of normal and highly lignified cell walls, and the middle lamellae. Important measures to describe the progress of the lignification are lumen area, cell wall area, cell wall thickness and the percentage of highly lignified area in the cell wall.

In Papers I and II, we proposed methods to segment compression wood fiber cross-sections into lumen (L), normal lignified (NL) and highly lignified (HL) cell walls and middle lamellae (ML). The purpose is to determine proper measurements of the fibers' attributes to be able to achieve a greater understanding of compression wood in general.

4.2 Segmentations

Our first approach, which we published in Paper I, works well for compression wood cells with distinct regions for L, NL and HL. But when these requirements are not satisfied, for example, when HL is partly or completely absent, this algorithm has problems. Therefore, we developed it further and published a more general approach in Paper II.

Both methods have the same work flow: We start with detecting all cell lumens. Next, we apply snakes to refine the boundary of the each lumen and to find the outer contour of NL. Finally, the cell boundary is detected. The implementations of this last step differ for the two proposed methods.

Lumen delineation

Since the lumen areas are much darker than the rest of the image, a suitable threshold can be used to extract them. But the non-uniform illumination and other imaging issues, such as non-perpendicular cut of the sample, make the segmentation challenging. We developed a rather complicated approach in Paper I, using the grey value of regions (lumens) that are enclosed by other regions (cell walls). We applied a simpler bias correction in Paper II followed by global thresholding. The delineation of the contour is then refined in both methods by using a snake.

Boundary between NL and HL

To find the boundary between the regions NL and HL (or the outer cell boundary, if HL does not exist), we use a two-stage snakes, as described in Section 2.2. To move the snake away from the contour of the lumen, we apply the inverted external force, which means high energy occur close to edges and low energy in regions with low gradient. With the help of a small balloon force, the snake settles approximately half way between the boundaries. Next, the original external force and again a small balloon force are applied and the snake moves on, attracted to the desired boundary.

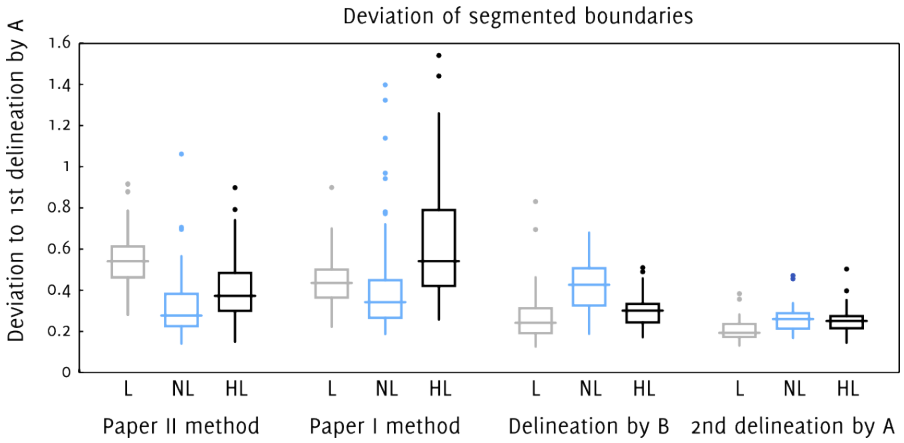

Figure 16: Deviation of automatic and manually determined boundaries to first delineation by expert A.

Cell wall boundary

In the end, the outer contour of the cell wall is delineated. In Paper I, we applied another two-stage snake, which works well for cells that have a clearly distinct HL. But for cells where HL is partly or completely missing, the snake overshoots the cell wall boundary, as described in Section 2.2, and the quality of the resulting segmentation in this area is diminished. Hence, we improved this step in Paper II by dividing the area outside NL into HL and ML. To separate HL and ML, we apply a line detector that finds a fine dark line around the cell outline. Since we are not longer presuming that HL exists, it is less probable that the improved segmentation fails.

4.3 Evaluation and discussion

We compared the proposed methods with manual segmentations using fluorescence microscopy images of compression wood cross-sections of Scots pine. We had two experts, here referred to as A and B, to delineate a set of cells. Expert A segmented a subset of the cells twice, to estimate intra-subject variance. Fig. 16 shows the deviation of the boundaries found by expert A in the first delineation. For the formula to determine the deviation, see Paper II.

The segmentation boundaries of the lumens differed most for the method published in Paper II. This is because of a systematic error, which is caused by a disagreement where to locate the boundary of the lumen. The algorithm sets the segmentation line where the gradient is highest, whereas

the experts define it slightly outside this location. That the method from Paper I came to a better matching result is due to a blurring operation that coincidentally shifted the segmentation boundaries in the right direction.

In Fig. 16, you can see that the delineation of the outer cell boundary improved for the method proposed in Paper II. This is because the two-stage snake from Paper I is more likely to overshoot when applied on this boundary. Hence, the improved method, using line detection, is more suitable for this problem and results in a more accurate delineation of the cell contour.

We showed that the two-stage snake is useful to find successively two boundaries that are both transitions from dark to bright. But by applying this method to the outer cell contour, we also demonstrated that the method has difficulties when the applied energies are not distinct enough. Here, the transition between the cell wall and ML is in some places very fuzzy. Hence, the gradient magnitude in these areas is not great enough to create a force that can hold back the snake, which leads to an overestimation of the HL regions.

We found that the parameters of snakes are not very intuitive to choose. But once we had determined optimal parameters, we were able to apply the method with the same parameters to images of different wood species and obtain a suitable segmentation result. The only value we needed to change was μ in connection with the GVF.

5 Application: Confocal images of human corneal endothelium

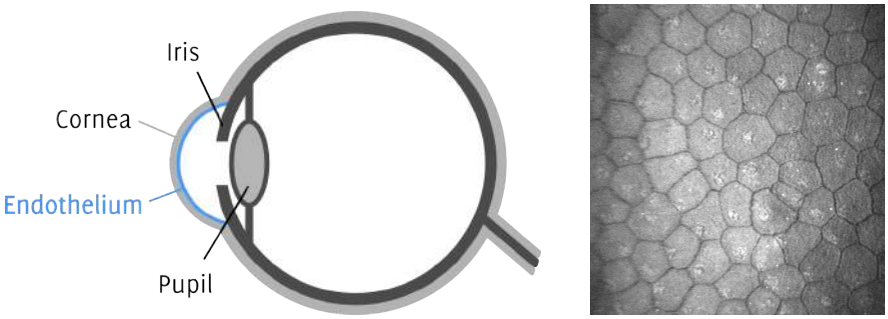

Figure 17: Illustration of eye (left) and example of an in vivo confocal microscopy image of corneal endothelium (right).

5.1 Background

The cornea is the transparent front layer of the eye covering the pupil and the iris.The endothelium of the cornea is a single layer of cells on the inner surface of the cornea. Its main purpose is to regulate fluid and solid transport between cornea and anterior chamber.

The endothilium is built of hexagonal cells that are developed prenatally. When endothelial cells die, the surrounding cells get thin and elongate to fill the empty space. Hence, the endothelial cell density decreases during life time. Different factors, such as inflammation or intraocular surgery, can speed up the endothelial cell loss even more. The transparency of the cornea gets affected when the endothelial cell density becomes too low.

In Paper VII, we present a parameter-free method to estimate the endothelial cell density in in vivo confocal microscopy images using spatial frequency analysis. With this method, we obtain estimates that are more precise than those of any other published method.

To determine other morphometric quantities, like endothelial cell size and shape, a complete segmentation of the image is necessary. In Paper VII, we present a completely automatic segmentation method based on the robust stochastic watershed (RSW), described in Section 3.6.

In Section 5.4, we extend the proposed method by using the fast evaluation of the robust stochastic watershed (FRSW), described in Section 3.7, and compare the results.

5.2 Cell density estimation

Due to the rather regular formation of the endothelial cells, we can use frequency analysis to estimate an average cell size and an approximate cell density. For this, we first compute the Fourier transform of the image. Then we remove the central peak using dilation by reconstruction [40], and finally, determine the radial mean $\mathscr{F}(f)$ of the result with

$$\mathscr{F}(f) = \frac{1}{2\pi} \int_0^{2\pi} |\mathscr{F}_{\text{rec}}(f, \theta)| \, d\theta, \tag{15}$$

where $\mathscr{F}_{\text{rec}}(f, \theta)$ is the Fourier transform of the image in polar form after the dilation with reconstruction operation, f is the radial frequency and θ is the angle.

The maximum of the radial mean corresponds to the *characteristic frequency* f^*, which relates to the most common cell width. The cell density δ can now be determined by

$$\delta = \frac{1}{\alpha} f^{*2}, \tag{16}$$

where α is a value that depends on the shape and regularity of the cells.

Fig. 18, we illustrate how f^* and consequently α change, when regarding images with different shaped patterns. In both the images (a) and (c) the side-to-side length of the cells is 25 pixels. In the frequency spectra (b) and (d), we determine $f^* = 14$ pixel for the square pattern and $f^* = 16$ pixel for the hexagonal pattern. Accordingly, $\alpha = 1$ for the square pattern and $\alpha \approx 1.13$ for the hexagonal pattern.

Publications by other authors [8, 17, 41] did not consider the shape of the cells when estimating the cell density, and therefore made the implicit assumption $\alpha = 1$. This implicit assumption did not influence the result much, because the actual value for α concerning endothelial cell images of this kind is close to 1. At first, this value seems unexpected, since it is the same as the square pattern and not as the hexagonal pattern in Fig. 18. But the imaged endothelial cells are not as perfectly regular as the cells in Fig. 18, which is the reason for the different value for α.

In Paper VII, we show that our method achieves more precise results than any of the previous published methods, irrespective of including α in the calculations.

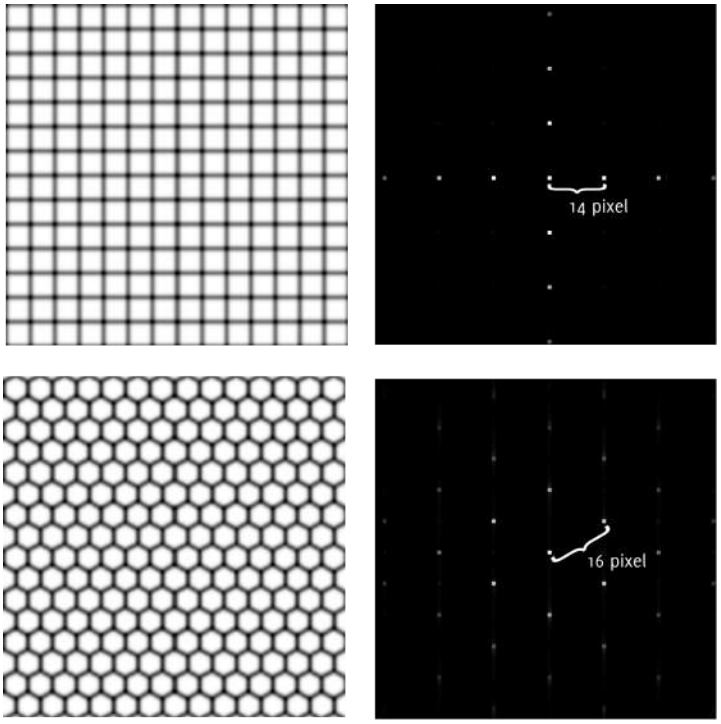

Figure 18: Images with square and hexagonal cell patterns with 25 pixels side-to-side length (left column), and magnitude of central region of the respective frequency spectra (right column).

5.3 Cell segmentation

The endothelial cell segmentation algorithm, proposed in Paper VII, is based on the RSW. We altered the method slightly from the version explained in Section 3.6: before applying the final step, which is the watershed segmentation using H-minima, we smoothed the PDF with a Gaussian blur. This operation complements the H-minima transform in simplifying the image. We made the σ of the smoothing operation dependent on the estimated cell density by defining a parameter $k_\sigma = \sigma f^*$.

Even though the smoothing has similar effects as the H-minima transform, we chose to apply both methods successively, because this combination gave us the best results. However, the smoothing has the disadvantage of slightly displacing the segmentation lines. To correct for this, we apply a seeded watershed on the original image using a slightly shrunk version of the found cells as seeds. This step adjusts the segmentation boundaries to better fit the structure of the original image.

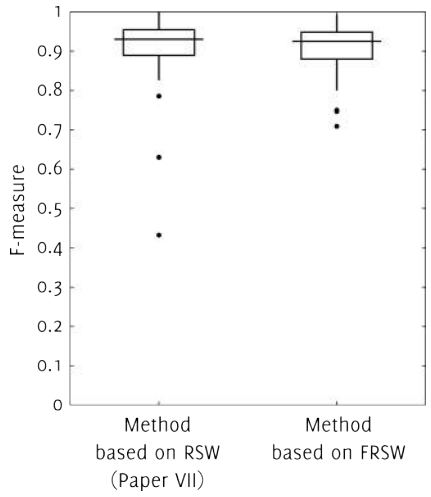

Figure 19: F-measure of the segmentations using the method based on RSW (proposed in Paper VII) and the new method based on FRSW.

For the evaluation in Paper VII, we had a data set available that included 52 confocal images of corneal endothelium. The images were acquired in 23 patients within the first year after endothelial keratoplasty (replacing corneal endothelium with donor tissue). Additionally, we had the morphometric measures of a semi-automatic and a fully automatic segmentation for the region of interest in each image. These measures were obtained by using the commercial software NAVIS (Nidtek Technologies SRL, Padova, Italy). We consider the measures determined with the semi-automated segmentation as ground truth.

For evaluating our approach, we set $M = 100$ and $N = A_I\delta$, where A_I is the size of the image and δ the estimated cell density. We trained the rest of the parameters using the leave-one-method.

The proposed method outperforms the automatic segmentation approach by NAVIS concerning all three determined morphometric measures: cell density, polymegatism (cell size variability) and pleomorphism (cell shape variation).

5.4 Alternative segmentation algorithm

After preparing Paper VII, we developed the FRSW, a fast evaluation of the RSW. In this section, we alter the proposed segmentation method by exchanging the RSW with the FRSW. We apply the new approach to the data set and compare the performance to the method introduced in Paper VII.

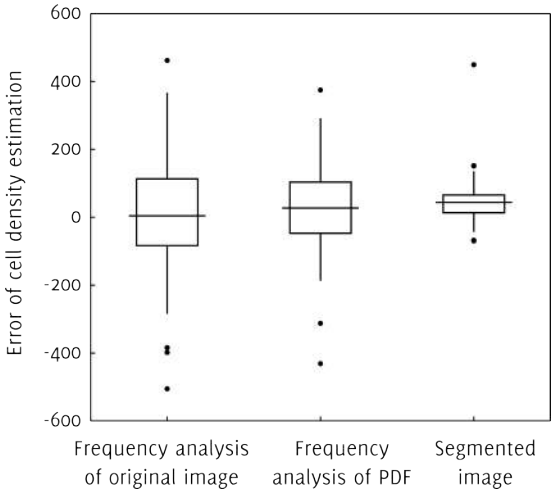

Figure 20: Error of cell density estimation using frequency analysis of the original image and of the PDF, and by counting cells in segmented image.

First, we determine the parameter for the new method, based on FRSW. For this, we set $M_f = 3$, and $N = A_I \delta$, as in Paper VII, and trained the other parameters with the leave-one-out method.

Finally, we calculate the F-measures for the final segmentation with the new approach and compare it to the F-measures of the segmentations of the method proposed in Paper VII, see Fig. 19. To calculate the F-measure, we only used the manually set centroids of the cells and the outline of the regions of interest. The exact procedure to determine this value is described in Paper VII.

The overall quality of the resulting segmentations is the same, as they produce very similar F-measures. The morphometric measures estimated from the segmentations (not shown) are similar.

5.5 Discussion

We showed that the RSW and the FRSW are well suited to segment corneal endothilium. The methods outperform the specialized commercial software NAVIS.

Additionally, we proposed in Paper VII a parameter-free approach to estimate the endothelial cell density.

We observed that the measure of the cell density becomes slightly more accurate when using the PDF (during the segmentation process) for the frequency analysis. But this requires the determination of the noise strength

s. In Fig. 20, we show the error of the estimated cell density determined with frequency analysis of the original image (proposed in Paper VII), of the PDF, and by counting the cells in the automatic segmentation. We see that the best cell density estimation is determined using the segmented image.

6 Conclusions and future perspective

6.1 Contributions

This thesis focuses on two different image segmentation approaches: snakes and stochastic watershed. Improvements of these methods were proposed to suit a specific problem or to make the approach more efficient.

In the next sections, we discuss the main contributions of this thesis.

The two-stage snakes (Paper I)

We developed an approach using snakes to delineate concentric regions, when their contours have the same polarity. This means that with our method it is possible to successively find two contours that are both transitions from for example a darker to a lighter region.
The two-stage snake approach works well as long one can produce a complementary energy that has its minimum between the two desired contours. Otherwise the snake overshoots and does not delineate the targeted boundary. Possible applications for this approach might be the segmentation of nucleus and corresponding cell wall of cells, or iris detection.

Tailored image analysis solutions for compression wood cross-section (Papers I and II)

We developed the first automatic segmentation method for fluorescence microscopy images of compression wood cross-section. For this, we successfully applied the two-stage snake to delineate the contours of the different regions in the cells. From the resulting segmentation, we could determine measurements that match the ground truth.
Another method to delineate these type of cells might be multi-layer level sets [11]. Alternatively, one could change the representation of the cells and regard their radial profiles. Then it is possible to use for example automatic methods to segment retinal layers in spectral domain optical coherence tomography images [10].

Analysis and improvements of the concept of stochastic watershed (Papers III, IV, V and VI)

After studying the strengths and weaknesses of the original stochastic watershed approach, we developed several versions to make the method more efficient. The ESW determines the exact PDF in a fraction of the computational time, and the RSW suppresses repeatedly found false boundaries by slightly altering the input image for each realization. Finally, we developed the FRSW, a hybrid approach combining the ESW and the RSW.

This method is both fast and improves the quality of the segmentation results compared to the original approach.
One weakness of the stochastic watershed methods is the problem to detect small regions. In the PDFs of all stochastic watershed approaches, the boundaries around small regions are weaker than around large regions. This is because it is less probable that random seed points are located in smaller regions. Using the RSW, it is even possible that structures that are smaller than the distance between the grid points, that is the distance between the seed points, is not detected at all by the algorithm.
Recently, a multi-scale approach of bagging stochastic watershed was proposed by Franchi and Angulo [18]. During their experiments, their proposed method and the RSW produced segmentation results with similar F-measures, which means that they performed equally good.
Applications for the proposed stochastic watershed methods are all task where the desired objects are distinguishable from other object or the background due to edges or lines in the gray value image. Examples for these tasks are the segmentation of cells in microscopy images or the segmentation of forest, fields and houses in satellite images.

Tailored image analysis solution for corneal endothelium (Paper VII)

We implemented a fully automatic segmentation algorithm using our stochastic watershed approaches. The measures determined with our algorithm agree with the ground truth better than the measures obtained with the commercial software.
There are several automatic approaches to segment in vitro corneal endothelium images [42, 45]. But due to their better image quality, these methods are hardly applicable to in vivo images we concentrated on. Gavet and Pinoli [19] and Vincent and Masters [47] proposed applicable automatic segmentation methods for in vivo images. In Paper VII, we show that our approach outperform these methods.

Correction of model for corneal endothelium cell density estimation (Paper VII)

During our work on microscopy images of corneal endothelium, we discovered a systematic error in all previous published methods based on frequency analysis. The model used to calculate the cell density was missing the incorporation of the cell shape. For this, we introduced the factor α, which adjusts for the appearance of the cell pattern.

6.2 Outlook

6.2.1 Future work

We developed image segmentation algorithms for fluorescence microscopy images of compression wood cross-sections and confocal microscopy images of corneal endothelium. Additionally, we proposed procedures to determine specific parameters of the imaged cells.

It would be interesting to define additional measurements in the future. For example, one could investigate if the area of the middle lamellae in compression wood cross-section correlates to the progress of the lignification or to any physical attributes.

We also noticed that some of the morphometric measurements used to describe the condition of corneal endothelium are more sensitive to errors in the segmentation. The pleomorphism is defined as the percentage of cells that are adjacent to exactly six neighboring cells. If, for example, one individual cell is over-segmented, not only the measure of this single cell is affected, but the measures of the entire neighborhood of this cell. Therefore, it can be more suitable to use measures that can be determined independently for each cell, as for example cell compactness or elongation.

Further, we see a great potential in the concept of the stochastic watershed. With the FRSW, we developed an approach that produces better segmentation results in less time, compared to the original method. The FRSW needs to be investigated further. In the next two sections, we discuss that the choice of parameter p in the power mean M_p influences the outcome of the segmentation, and possibilities to make the result of the FRSW deterministic.

6.2.2 Influence of parameter p

The FRSW creates M_f intermediate PDFs, which are merged to form the final PDF. Since it is less probable to find the same false boundaries repeatedly, single occurrences of high values in the intermediate PDFs are suppressed. Hence, it is beneficial to apply a measure that favors small values, such as the power mean M_p for $p < 1$.

This measure is a non-linear transformation that places more weight on small input values. The degree of the weighting is determined by parameter p. The smaller p, the more emphasis on low values, see Fig. 21. The power mean $M_{-\infty}$ corresponds to the minimum.

In our experiments, we found M_0 to perform best. But we consider that it is more likely that the optimal p depends on the application. The probability of the occurrence of false boundaries is different for every problem. Therefore, the level of how much they need to be suppressed to find an op-

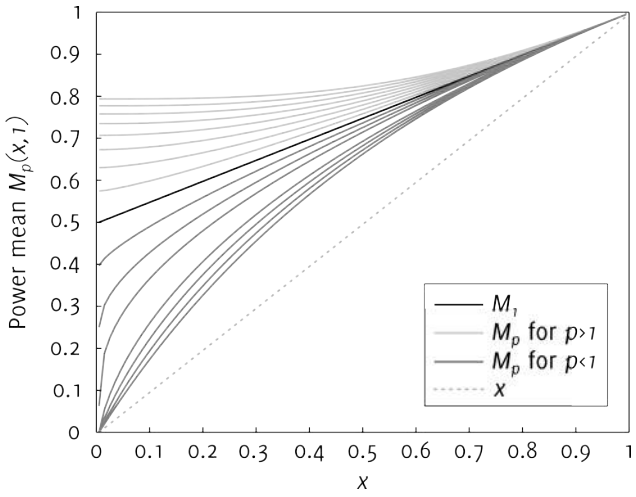

Figure 21: Power mean $M_p(x,1)$ for different values of p. The black line shows M_1, the light gray lines illustrate M_p for $p = 1.25, 1.5, ...3$ and the dark gray lines show M_p for $p = 0.75, 0.5, ... - 1$.

timal segmentation can vary. Possibly, the optimal p also depends on the number of realizations. So it needs to be verified how sensitive the FRSW is to parameter p.

6.2.3 Deterministic version of FRSW

One advantage of most segmentation algorithms over a human expert is that their results are deterministic. When a human is delineating an object manually, he or she will be unlikely to draw exactly the same outline again, when repeating the experiment. We say that most of segmentation algorithms have a high precision (no occurrence of random errors).

However, methods based on Monte Carlo simulations, as SW, RSW and FRSW, are not deterministic. They rely on repeated random sampling to achieve an approximation of the exact result. This means that when applying an algorithm involving Monte Carlo simulations twice on the same image, we obtain results that slightly differ from each other. That might be undesirable for some applications.

With the ESW, we can determine the exact result by involving the probability of the sampling in the calculations. Thus, we could not only speed up the process, but also make the results deterministic.

In the FRSW, we use the ESW together with the concept of adding noise to the image to overcome false boundaries. This introduces randomness to the calculations of the ESW, which leads to the results being not determin-

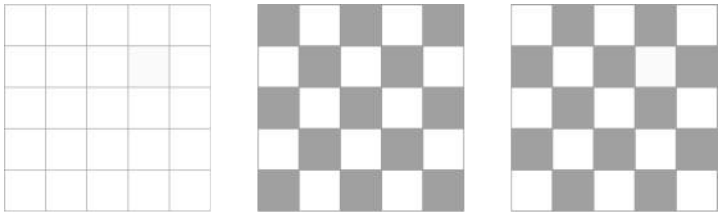

Figure 22: Example for $M_f = 3$. Offset can be added to the gray pixels during the first, second and third realization of ESW.

istic.

As described before in Section 3.6, false boundaries occur because the SW is not very sensitive to the placement of seed points. This means that the order in which the pixels are added to the emerging regions is always the same. To overcome the problem, this order needs to be slightly altered to not repeatedly enhance the same false boundaries. In the RSW, we solve this problem by adding a small amount of noise to the image before each realization of seeded watershed. In FRSW, we determine the exact PDF of three different images: the original image and two variations of the image with a small amount of added noise.

Here, we discuss some ideas to make the results of the our segmentation approach deterministic and at the same time prevent the algorithm from finding false boundaries.

Offset instead of noise: By adding noise to the image, we slightly change the gray level values of the pixels. Instead of choosing random values, we could select a constant offset.

When choosing $M_f = 3$, as suggested in Paper VI, we can realize our idea as follows: The first realization would be performed with the original image. For the second realization, an offset would be added to every second pixel in the image and for the third realization, the offset would be added to pixels of the original image that were not altered in the previous realization, see Fig 22.

We assume that the optimal value for the offset lies around half of the optimal noise strength.

Alternative MSTs: Another approach is to literally change the order of the pixels. When applying the ESW, we transform the image into a graph representation. To determine the MST, all edges must be stored according to their weight in a sorted list. The order of this list corresponds to the order in which the pixels are added to the evolving regions. Hence, when

rearranging the edges in the list, a different MST is created. The task lies in modifying the order in such a way that false boundaries are suppressed, but at the same time salient boundaries repeatedly found. This corresponds to finding the optimal noise level in the FRSW.

To alter the order of the sorted list can be realized in different ways. Simple examples would be to switch the position of two successive edges or to shift every second pixel by a fixed number of positions.

Additionally to making the results deterministic, we might be able to gain even a slight speed-up in our implementation. Because transforming the input image into graph representation and sorting the edges need to be performed only once, instead of M_f times.

Straehle et al. [44] proposed a method to create almost-MSTs that could be also useful for this approach.

Sammanfattning (Summary in Swedish)

Syftet med digital bildanalys är att utvinna meningsfull information från digitala bilder. Det finns många applikationer. Bildanalys kan till exempel användas för att mäta avstånd och ytor i satellitbilder, att känna igen ansikten i övervakningsbilder och att upptäcka tumörer i medicinska bilder. Eftersom tillämpningarna är så olika är det omöjligt att skapa en universell bildanalysmetod. Vi måste istället skapa en ny, lämplig algoritm för varje nytt problem.

För att utvinna information ur en bild måste vi först definiera intressanta regioner eller objekt. För att kunna göra detta måste bilden segmenteras, vilket betyder att dela upp den i regioner med homogent innehåll. Segmenteringen kan göras manuellt, halvautomatiskt eller helt automatiskt, beroende på omständigheterna. Eftersom manuell segmentering är tröttsamt och dessutom subjektivt och därför riskerar introducera felaktigheter föredrar vi ofta automatiska metoder. Helautomatiska metoder är mest effektiva, eftersom halvautomatisk segmentering kräver insatser en människa, som dessutom kanske måste vara expert inom tillämpningsområdet. Men detta innebär att helautomatiska metoder själva måste kunna skapa pålitliga resultat av hög kvalité.

I denna avhandling föreslår vi förbättrade versioner av två olika existerande segmenteringsmetoder: ormar (“snakes”) och stokastisk vattendelare (“stochastic watershed”).

Ormar används när den ungefärliga formen av och läget för objektets kontur är känd. Processen kan beskrivas som ett energiminimeringsproblem: när ormen initialiserats utsätts den för olika krafter som iterativt flyttar den mot den önskade konturen. Slutligen stannar den över konturen, eftersom de pålagda krafterna där når ett energiminimum.

Ormar misslyckas (till exempel) när ett antal koncentriska konturer med liknande form ska upptäckas efter varandra, eftersom det är omöjligt för ormen att förflytta sig från en kontur till nästa. Vi föreslår en tvåstegsprocess (“two-stage snake”) för att lösa problemet: när en orm hittat den första konturen skapas en temporär, omvänd orm som repelleras av konturer och därmed trycks bort från den första konturen för att sedan kunna attraheras av nästa kontur.

För att illustrera den föreslagna metoden använde vi tvåstegsormen för att segmentera fluorescensmikroskopibilder av tjurvedsceller. För denna uppgift fanns tidigare inga fungerande segmenteringsmetoder. Vi visade att vår metod kan användas i stället för manuell segmentering, eftersom våra automatiskt framtagna resultat är starkt korrelerade till segmenteringar som gjorts manuellt av experter.

I avhandlingen diskuterar vi även stokastisk vattendelare. Metoden skapar en täthetsfunktion som visar de mest betydelsefulla konturerna i bilden, där sannolikheten för en kontur är stor. Stokastisk vattendelare är speciellt lämpligt för strukturer som består av regioner av liknande storlek och där det ungefärliga antalet regioner är känt.

Originalversionen av statistisk vattendelare behövde förbättras vad det gäller beräkningstid och undertryckande av falska konturer. Vi föreslår en metod för exakt beräkning av stokastisk vattendelare ("exact evaluation of stochastic watershed", ESW) och robust stokastisk vattendelare ("robust stochastic watershed", RSW), som löser respektive problem. Med ESW kan vi beräkna den exakta täthetsfunktionen utan Monte Carlo-simulering. I stället använder vi grafteori. Den algoritm vi utvecklat är betydligt snabbare än originalversionen. RSW använder brus för att störa ut de svaga konturer som konsekvent uppstår i större områden. Den förbättrar därför resultatet i bilder där objekten är olika stora. För att ta vara på fördelarna med de båda nya metoderna sammanslog vi dem till snabb robust stokastisk vattendelare ("fast stochastic watershed", FRSW). Den nya hybridmetoden använder några få ESW-beräkningar på en bild där brus adderats på liknande sätt som i RSW. Detta förkortar beräkningstiden signifikant för den robusta metoden.

För att illustrera de nya metoderna använde vi RSW och FRSW för att segmentera konfokalmikroskopibilder av hornhinnans endoltel som tagits in vivo. Våra metoder ger bättre resultat än den automatiska segmenteringsmetod som finns i den kommersiella programvaran NAVIS. Vi noterade också att alla de tidigare publicerade metoder som använder frekvensanalys för att uppskatta tätheten hos hornhinnas endotel bortser från cellernas form. I avhandlingen introducerar vi en parameter som kompenserar för cellformen och dessutom en ny metod som ger bättre resultat än tidigare metoder.

Denna avhandling ger nya bidrag till området automatisk bildsegmentering. Vi har utvecklat metoder som förbättrar och rättar existerande algoritmer. Vi föreslår att dessa tillämpas vid automatisk analys av mikroskopibilder inom trävetenskap och oftalmologi.

Zusammenfassung (Summary in German)

Digitale Bildanalyse wird dazu genutzt, um aussagekräftige Informationen aus digitalen Bildern zu gewinnen. Das Anwendungsgebiet ist vielfältig. Es reicht von Landvermessung mittels Satellitenbildern, über Gesichterkennung in Überwachungsbildern, bis hin zur Krankheitsdiagnose durch die Analyse von medizinischen Bildern. Da sich die Anforderungen der Anwendungen stark von einander unterscheiden, muss das verwendete Verfahren für jedes Probelm neu zugeschitten werden. Eine allgemein anwendbare Bildanalysemethode gibt es daher nicht.

Um Information aus dem Bild zu extrahieren, müssen die dafür interessanten Bereiche oder Objekte im Bild gefunden werden. Dazu wird das Bild segmentiert, d.h. in inhaltlich zusammenhängende Regionen unterteilt. Je nach Möglichkeit wird die Segmentierung manuell, halbautomatisch oder vollautomatisch durchgeführt. Da manuelle Segmentierung eine langwierige Prozedur ist, die dazu noch subjektiv und fehleranfällig ist, werden oft automatische Verfahren bevorzugt. Während halbautomatische Verfahren noch die Eingabe des Nutzers brauchen, der eventuell ein Experte im Anwendungsgebiet sein muss, ist eine vollautomatische Segmetierung effektiver. Sie wird ohne menschliches Einwirken erstellt und spart daher Arbeitszeit und Aufwand. Aber gerade deswegen ist es erforderlich, dass vollautomatische Verfahren eigenständig zuverlässige und qualitativ hochwertige Ergebnisse liefern.

Die vorliegende Arbeit leistet einen Beitrag zu der Verbesserung der zwei bestehenden vollautomatischen Segmetierungsmethoden *Snakes* (z.Dt. Schlangen) und *Stochastic Watershed* (z.Dt. stochastische Wasserscheidentransformation).

Snakes werden angewandt um den Umriss eines Objektes zu finden, dessen ungefähre Form und Position bekannt ist. Sie werden als (geschlossene) Konturen dargestellt und zu Beginn möglichst nah der erwarteten Zielposition platziert. Nach der Initialisierung werden Kräfte ausgeübt, die die Snake über das Bild bewergen. Diese angewandten Kräfte legen die bevorzugte Silhouette der Snake fest und führen sie in Richtung des Umrisses des Zielobjekts. Dieses Verfahren ist ein Energieminimierungsverfahren, das sein Optimum erreicht, wenn sich die angreifenden Kräfte ausgleichen und die Snake somit zum Stillstand kommt.

Das Konzept der Snakes war bisher nicht anwendbar, wenn zwei konzentrische Konturen mit ähnlichen Eigenschaften nacheinander gefunden werden sollten. Der Grund dafür ist, dass es für die angreifenden Kräfte unmöglich ist die Snake von der ersten Kontur wegzustoßen und gleichzeitig ein Kräftegleichgewicht an der zweiten Kontur zu erreichen. In dieser Ar-

beit wird die *Two-Stage Snake* (z.Dt Zwei-Stufen-Schlange) vorgestellt, die dieses Problem mittels eines Zwischenschritts überwindet. Zunächst werden Kräfte angelegt, so dass sich die Snake von der ersten Kontur wegbewegt und auf ungefähr halbem Weg bis zur zweiten Kontur zum Stehen kommt. Daraufhin werden neue Kräfte angelegt, die bewirken, dass die Snake von der zweiten Kontur angezogen wird und dort ihre Endposition erreicht. Das Konzept der Two-Stage Snake wird in dieser Arbeit an Fluoreszenzmikroskopiebildern von Druckholz illustriert, für die es bisher keine vollautomatische Segmentierungsmethode gab.

Weiterhin wird in dieser Arbeit das Stochastic-Watershed-Verfahren diskutiert und weiterentwickelt. Dieses Verfahren ist ursprünglich eine Monte-Carlo-Methode, die eine Wahrscheinlichkeitsdichtefunktion der prominentesten Konturen in einem Bild berechnet. Anhand dieser Funktion kann daraufhin eine Segmentierung erstellt werden. Das Stochastic-Watershed-Verfahren ist besonders für Bilder geeignet, die aus etwa gleichgroßen Bereichen bestehen, dessen ungefähre Anzahl bekannt ist.

Das Stochastic-Watershed-Verfahren hat Verbesserungspotential bezüglich Laufzeit und Qualität der Segmentierung. In dieser Arbeit werden die Methoden *Exact Evaluation of Stochastic Watershed* (ESW, z.Dt. exakte Auswertung der stochastischen Wasserscheidentransformation) und *Robust Stochastic Watershed* (RSW, z.Dt. robuste stochastische Wasserscheidentransformation) vorgestellt, die jeweils eine der genannen Schwachstellen angehen und das Verfahren daraufhin verbessern.

Mit ESW wird die exakte Wahrscheinlichkeitsdichtefunktion mittels Graphtheorie berechnet ohne dabei Monte-Carlo-Simulation anzuwenden. Die hier vorgestellte Methode ESW ist um ein Vielfaches schneller als die Originalmethode.

Das Stochastic-Watershed-Verfahren neigt dazu große Regionen zu teilen, auch wenn es keine ersichtlichen Kanten gibt. Damit während der Monte-Carlo-Simulation nicht wiederholt die gleichen falschen Segmentierungslinien gefunden werden, wird bei RSW ein leichtes Rauschsignal zum Bild hinzugefügt. Somit wird die Qualität der Segmentierung verbessert, da das Finden von falschen Linien in der endgültigen Segmentierung verhindert wird. Diese neue Version des Stochastic-Watershed-Verfahrens ist nun auch für Bilder anwendbar, die verschieden große Objekte darstellen.

Um die Vorteile der beiden neuen Methoden zu nutzen, wird in dieser Arbeit die Methode *Fast Robust Stochastic Watershed* (FRSW, z.Dt. schnelle robuste Stochastische Wasserscheidentransformation) vorgestellt. Sie verwendet wenige Wiederholungen des ESW, bei denen das Bild wie beim RSW jedesmal durch Rauschen verändert wird. Die Ergebnisse von RSW und

FRSW sind qualitativ gleichwertig, allerdings ist FRSW um ein Vielfaches schneller als RSW.

Schließlich werden die Methoden RSW und FRSW an in vivo Konfokalmikroskopiebildern der Endothelzellschicht von menschlicher Hornhaut illustriert. Diese Methoden erzeugen qualitativ bessere Ergebisse als die kommerziell vertriebene Software NAVIS.

Weiterhin wird beschrieben, wie sämtliche bisher veröffentlichten Methoden zur Bestimmung der Zelldichte in Endothelzellschichten die Zellform außer Acht lassen. In dieser Arbeit wird daher ein Parameter eingeführt, der die Zellform kompensiert. Zusätzlich wird ein neues Verfahren vorgestellt, das qualitativ bessere Ergebnisse liefert als die bisher veröffentlichten Methoden.

Die vorliegende Arbeit stellt einen Beitrag zum Themengebiet automatische Bildsegmentierung dar. Im Rahmen dieser Arbeit wurden Methoden entwickelt, die bisherige Ansätze verbessern und korrigieren. Weiterhin wurden Verfahren zur automatische Analyse von Mikroskopiebildern in den Fachgebieten Holzwissenschaft und Augenheitkunde vorgestellt.

Acknowledgments

This thesis is the result of the research conducted at the Centre for Image Analysis, Uppsala, Sweden. I would like to express my sincere gratitude and appreciation toward all those people who have contributed, directly or indirectly, to this thesis.

- Cris Luengo, my main supervisor, for always having your door open and incredible patience. Thank you for believing in me, supporting me, and easing a lot of workload during my most stressful period.
- Gunilla Borgefors, my co-supervisor, for sharing your great experience, teaching us how to work scientifically and for understanding.
- Stig Bardage for supervising the first period of my PhD studies. Thank you and Geoffrey Daniel for fruitful collaboration.
- Filip Malmberg for being inspiring and always happy to answer my questions.
- Lena Nordström for taking care of everything. You are the heart of CBA!
- Olle Eriksson for never letting me have computer problems. CBA is not the same without you.
- Ingela Nyström for understanding and making things possible.
- Bernd Rieger, his colleagues at the Quantitative Imaging Group in Delft, Koen Vermeer and his colleagues at the Eye Hospital in Rotterdam for making my stay in the Netherlands interesting, fruitful and fun.
- Cris, Gunilla, Filip and Erik for proof-reading this thesis.
- Gustaf for always being there for me. You are a true friend.
- Patrik for making kroppkakor for me, Erik for your brass fasteners (they did a great job!), Hamid for encouraging me to apply for a PhD position at CBA, Amin for taking me out dancing and partying, Elisabeth for telling me about "duktig flicka syndromet", Pontus for spreading your good mood, Andreas for being open and sharing, Vlad for asking me for advice from time to time, Lennart for always being helpful and kind, Ida-Maria for caring and hugging, Kristin for marrying in an orange dress, Milan for that awesome Duplo fire truck,

Khalid for wasting time, and all my other friends and colleagues at Vi2 for friendly words and smiles on your faces.

- Anna Karlgren and Kerstin Malm for friendship, compassion and encouragement. Thanks for sharing your experiences that night at Katalin. It helped me a lot.
- Katarina Lundvall for adventurous trips and constantly believing in me. Your enthusiasm and positive thinking is encouraging.
- Erica Hamilton for introducing mindfulness to my daily life and sharing the challenges and joys of motherhood.
- Sandra Ander for listening, talking and having fun together, simply for being my friend.
- Joanna Meier and Lena Cederholm for welcoming me in your "bubble" and giving me the possibility to do something fun and useful.
- Jasna Mai, Saska Stanceska, Sugentha Thangaraja, Katharina Morr, Mimi Tenchio, Nicole Klameth and Petra Erdin for long-lasting friendship. Even though, we are far apart from each other, I still feel close connected to you.
- Lars-Åke, Maggan, BM, Conny, Anna, Jonas, Albin and Gustav for letting me be part of your family. I enjoy spending time with you!
- My deceased grandma Ellen. I am still learning from you.
- My mom Adelheid for always being there for me and trying to understand, my dad Heinz for letting me know how proud I make you, Andreas for being a great big brother and the idol of my childhood, Corina for being kind and easy to talk to, and Jannek and Leni for being sunshiny and having boundless energy.
- Emilian for your joy of life, your kindness, your enthusiasm, your strength, your patience, your thoughtfulness, your curiosity, your sharpness, your affection and your love. Thank you for the millions of kisses you gave me so far. You make my world a better place!
- Martin. Your creativity inspires me. Your view of the world lets me rethink. Your modesty impresses me. Your love fills me with joy. Thanks for being as you are. You make me happy. I am looking forward to our future life together. LUHM ♡

References

[1] J. Angulo and D. Jeulin, "Stochastic watershed segmentation", in *8th International Symposium on Mathematical Morphology (ISMM)*, 2007, pp. 265–276 (cit. on pp. 11, 20, 22).

[2] P. Arbelaez, M. Maire, C. Fowlkes and J. Malik, "Contour detection and hierarchical image segmentation", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 5, pp. 898–916, 2011 (cit. on p. 31).

[3] M. Axelsson, "Image analysis for volumetric characterisation of microstructure", PhD thesis, Centre for Image Analysis, Uppsala, Sweden, 2009 (cit. on p. 35).

[4] S. Bauer, L.-P. Nolte and M. Reyes, "Fully automatic segmentation of brain tumor images using support vector machine classification in combination with hierarchical conditional random field regularization", in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2011*, Springer, 2011, pp. 354–361 (cit. on p. 11).

[5] M. A. Bender and M. Farach-Colton, "The lca problem revisited", in *LATIN 2000: Theoretical Informatics*, Springer, 2000, pp. 88–94 (cit. on p. 27).

[6] S. Beucher and C. Lantuejoul, "Use of Watersheds in Contour Detection", in *International Workshop on Image Processing: Real-time Edge and Motion Detection/Estimation, Rennes, France.*, 1979 (cit. on p. 17).

[7] S. Beucher and F. Meyer, "The morphological approach to segmentation: the watershed transformation", in *Mathematical morphology in image processing*, 1993, pp. 433–481 (cit. on p. 18).

[8] C. Bucht, P. Söderberg and G. Manneberg, "Fully automated corneal endothelial morphometry of images captured by clinical specular microscopy", in *SPIE BiOS: Biomedical Optics*, International Society for Optics and Photonics, 2009, pp. 716 315–716 315 (cit. on p. 40).

[9] J. Canny, "A computational approach to edge detection", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, no. 6, pp. 679–698, 1986 (cit. on p. 11).

[10] S. J. Chiu, X. T. Li, P. Nicholas, C. A. Toth, J. A. Izatt and S. Farsiu, "Automatic segmentation of seven retinal layers in sdoct images congruent with expert manual segmentation", *Optics express*, vol. 18, no. 18, pp. 19 413–19 428, 2010 (cit. on p. 45).

[11] G. Chung and L. Vese, "Image segmentation using a multilayer level-set approach", *Computing and Visualization in Science*, vol. 12, no. 6, pp. 267–285, 2009 (cit. on p. 45).

[12] L. P. Coelho, A. Shariff and R. F. Murphy, "Nuclear segmentation in microscope cell images: a hand-segmented dataset and comparison of algorithms", in *Biomedical Imaging: From Nano to Macro, 2009. ISBI'09. IEEE International Symposium on*, IEEE, 2009, pp. 518–521 (cit. on p. 31).

[13] J. Cousty, G. Bertrand, L. Najman and M. Couprie, "Watershed cuts: thinnings, shortest path forests, and topological watersheds", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 5, pp. 925–939, 2010 (cit. on p. 19).

[14] L. A. Donaldson, "Lignification and lignin topochemistry—an ultrastructural view", *Phytochemistry*, vol. 57, no. 6, pp. 859–873, 2001 (cit. on p. 35).

[15] L. Donaldson, K. Radotić, A. Kalauzi, D. Djikanović and M. Jeremić, "Quantification of compression wood severity in tracheids of< i> pinus radiata</i> d. don using confocal fluorescence imaging and spectral deconvolution", *Journal of structural biology*, vol. 169, no. 1, pp. 106–115, 2010 (cit. on p. 35).

[16] M. Faessel and D. Jeulin, "Segmentation of 3d microtomographic images of granular materials with the stochastic watershed", *Journal of microscopy*, vol. 239, pp. 17–31, 2010 (cit. on p. 23).

[17] M. Foracchia and A. Ruggeri, "Automatic estimation of endothelium cell density in donor corneas by means of fourier analysis", *Medical and Biological Engineering and Computing*, vol. 42, no. 5, pp. 725–731, 2004 (cit. on p. 40).

[18] G. Franchi and J. Angulo. (2015). Bagging stochastic watershed on natural color image segmentation, [Online]. Available: `https://hal.archives-ouvertes.fr/hal-01104256/` (cit. on p. 46).

[19] Y. Gavet and J.-C. Pinoli, "Visual perception based automatic recognition of cell mosaics in human corneal endothelium microscopy images", *Image Analysis & Stereology*, vol. 27, no. 1, pp. 53–61, 2011 (cit. on p. 46).

[20] W. Guoqiang and W. Dongxue, "Segmentation of brain mri image with gvf snake model", in *Pervasive Computing Signal Processing and Applications (PCSPA), 2010 First International Conference on*, 2010, pp. 711–714 (cit. on p. 12).

[21] D. Harel and R. E. Tarjan, "Fast algorithms for finding nearest common ancestors", *siam Journal on Computing*, vol. 13, no. 2, pp. 338–355, 1984 (cit. on p. 27).

[22] J. Immerkaer, "Fast noise variance estimation", *Computer Vision and Image Understanding*, vol. 64, no. 2, pp. 300–302, 1996 (cit. on p. 28).

[23] N. A. M. Isa, S. A. Salamah and U. K. Ngah, "Adaptive fuzzy moving k-means clustering algorithm for image segmentation", *Consumer Electronics, IEEE Transactions on*, vol. 55, no. 4, pp. 2145–2153, 2009 (cit. on p. 11).

[24] M. Kass, A. Witkin and D. Terzopoulos, "Snakes: active contour models", *International journal of computer vision*, vol. 1, no. 4, pp. 321–331, 1988 (cit. on pp. 11, 13).

[25] J. B. Kruskal, "On the shortest spanning subtree of a graph and the traveling salesman problem", *Proceedings of the American Mathematical society*, vol. 7, no. 1, pp. 48–50, 1956 (cit. on p. 19).

[26] T. M. Lillesand, R. W. Kiefer, J. W. Chipman *et al.*, *Remote sensing and image interpretation*. Ed. 5. John Wiley & Sons Ltd, 2004 (cit. on p. 11).

[27] F. Malmberg and C. L. Luengo Hendriks, "An efficient algorithm for evaluation of stochastic watersheds", *Pattern Recognition Letters*, vol. 47, no. 0, pp. 80–84, 2014, Advances in Mathematical Morphology (cit. on pp. 24, 25).

[28] F. Meyer, "Minimum spanning forests for morphological segmentation", in *Mathematical morphology and its applications to image processing*, Springer, 1994, pp. 77–84 (cit. on p. 19).

[29] F. Meyer and J. Stawiaski, "A stochastic evaluation of the contour strength", in *Proceedings of the 32nd DAGM conference on Pattern recognition*, 2010, pp. 513–522 (cit. on p. 24).

[30] M. Moëll, "Digital image analysis for wood fiber images", PhD thesis, Centre for Image Analysis, Uppsala, Sweden, 2001 (cit. on p. 35).

[31] R. Nock and F. Nielsen, "Statistical region merging", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 26, no. 11, pp. 1452–1458, 2004 (cit. on p. 11).

[32] K. Norell, "Automatic analysis of log end face images in the sawmill industry", PhD thesis, Centre for Image Analysis, Uppsala, Sweden, 2010 (cit. on p. 35).

[33] G. Noyel, J. Angulo and D. Jeulin, "Random germs and stochastic watershed for unsupervised multispectral image segmentation", in *Proceedings of the 11th international conference, KES 2007 and XVII Italian workshop on neural networks conference on Knowledge-based intelligent information and engineering systems: Part III*, 2007, pp. 17–24 (cit. on p. 12).

[34] G. Noyel, J. Angulo and D. Jeulin, "Classification-driven stochastic watershed: application to multispectral segmentation", in *CGIV 2008 – Fourth European Conference on Colour in Graphics, Imaging, and MCS/08 Vision 10th International Symposium on Multispectral Colour Science*, 2008, pp. 471–476 (cit. on p. 23).

[35] S. Osher and J. A. Sethian, "Fronts propagating with curvature dependent speed: algorithms based on hamilton-jacobi formulations", *Journal of Computational Physics*, vol. 79, no. 1, pp. 12–49, 1988 (cit. on p. 13).

[36] N. Otsu, "A threshold selection method from gray-level histograms", *Automatica*, vol. 11, no. 285-296, pp. 23–27, 1975 (cit. on p. 11).

[37] N. Paragios and R. Deriche, "Geodesic active contours and level sets for the detection and tracking of moving objects", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, no. 3, pp. 266–280, 2000 (cit. on p. 13).

[38] C. Plomion, G. Leprovost and A. Stokes, "Wood formation in trees", *Plant physiology*, vol. 127, no. 4, pp. 1513–1523, 2001 (cit. on p. 35).

[39] R. C. Prim, "Shortest connection networks and some generalizations", *Bell system technical journal*, vol. 36, no. 6, pp. 1389–1401, 1957 (cit. on p. 19).

[40] K. Robinson and P. F. Whelan, "Efficient morphological reconstruction: a downhill filter", *Pattern Recognition Letters*, vol. 25, no. 15, pp. 1759–1767, 2004 (cit. on p. 40).

[41] A Ruggeri, E Grisan and J Jaroszewski, "A new system for the automatic estimation of endothelial cell density in donor corneas", *British Journal of Ophthalmology*, vol. 89, no. 3, pp. 306–311, 2005 (cit. on p. 40).

[42] A. Ruggeri, F. Scarpa, M. De Luca, C. Meltendorf and J. Schroeter, "A system for the automatic estimation of morphometric parameters of corneal endothelium in alizarine red-stained images", *British Journal of Ophthalmology*, vol. 94, no. 5, pp. 643–647, 2010 (cit. on p. 46).

[43] J. Shi and J. Malik, "Normalized cuts and image segmentation", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, no. 8, pp. 888–905, 2000 (cit. on p. 11).

[44] C. Straehle, S. Peter, U. Köthe and F. A. Hamprecht, "K-smallest spanning tree segmentations", in *Pattern Recognition*, ser. Lecture Notes in Computer Science, vol. 8142, Springer Berlin Heidelberg, 2013, pp. 375–384 (cit. on p. 50).

[45] F. Tinena, P. Sobrevilla and E. Montseny, "On quality assessment of corneal endothelium and its possibility to be used for surgical corneal transplantation", in *Fuzzy Systems, 2009. FUZZ-IEEE 2009. IEEE International Conference on*, IEEE, 2009, pp. 1326–1331 (cit. on p. 46).

[46] L. Vincent and P. Soille, "Watersheds in digital spaces: an efficient algorithm based on immersion simulations", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 13, no. 6, pp. 583–598, 1991 (cit. on p. 18).

[47] L. M. Vincent and B. R. Masters, "Morphological image processing and network analysis of cornea endothelial cell images", vol. 1769, 1992, pp. 212–226 (cit. on p. 46).

[48] E. L. G. Wernersson, "Characterisation of wood-fibre-based materials using image analysis", PhD thesis, Centre for Image Analysis, Uppsala, Sweden, 2014 (cit. on p. 35).

[49] C. Xu and J. L. Prince, "Gradient vector flow: a new external force for snakes", in *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, IEEE, 1997, pp. 66–71 (cit. on p. 14).

[50] W. Zhao, R. Chellappa, P. J. Phillips and A. Rosenfeld, "Face recognition: a literature survey", *Acm Computing Surveys (CSUR)*, vol. 35, no. 4, pp. 399–458, 2003 (cit. on p. 11).

Paper I

Segmentation of highly lignified zones in wood fiber cross-sections

B. Selig, C. L. Luengo Hendriks, S. Bardage and G. Borgefors
in *Image Analysis*, ser. Lecture Notes in Computer Science, A.-B. Salberg, J.Y. Hardeberg and R. Jenssen, Eds., vol. 5575, Springer Berlin Heidelberg, pp. 369–378, 2009.

Abstract

Lignification of wood fibers has important consequences to the paper production, but its exact effects are not well understood. To correlate exact levels of lignin in wood fibers to their mechanical properties, lignin autofluorescence is imaged in wood fiber cross-sections. Highly lignified areas can be detected and related to the area of the whole cell wall. Presently these measurements are performed manually, which is tedious and expensive. In this paper a method is proposed to estimate the degree of lignification automatically. A multi-stage snake-based segmentation is applied on each cell separately. To make a preliminary evaluation we used an image which contained 17 complete cell cross-sections. This image was segmented both automatically and manually by an expert. There was a highly significant correlation between the two methods, although a systematic difference indicates a disagreement in the definition of the edges between the expert and the algorithm.

Paper II

Automatic measurement of compression wood cell attributes in fluorescence microscopy images

B. Selig, C. L. Luengo Hendriks, S. Bardage, G. Daniel and G. Borgefors
Journal of Microscopy, vol. 246, no. 3, pp. 298–308, 2012.

Abstract

This paper presents a new automated method for analyzing compression wood fibers in fluorescence microscopy. Abnormal wood known as compression wood is present in almost every softwood tree harvested. Compression wood fibers show a different cell wall morphology and chemistry compared to normal wood fibers, and their mechanical and physical characteristics are considered detrimental for both construction wood and pulp and paper purposes. Currently there is the need for improved methodologies for characterization of lignin distribution in wood cell walls, such as from compression wood fibers, that will allow for a better understanding of fiber mechanical properties. Traditionally, analysis of fluorescence microscopy images of fiber cross-sections has been done manually, which is time consuming and subjective. Here, we present an automatic method, using digital image analysis, that detects and delineates softwood fibers in fluorescence microscopy images, dividing them into cell lumen, normal and highly lignified areas. It also quantifies the different areas, as well as measures cell wall thickness. The method is evaluated by comparing the automatic with a manual delineation. While the boundaries between the various fiber wall regions are detected using the automatic method with precision similar to inter and intra expert variability, the position of the boundary between lumen and the cell wall has a systematic shift that can be corrected. Our method allows for transverse structural characterization of compression wood fibers, which may allow for improved understanding of the micro-mechanical modeling of wood and pulp fibers.

Paper III

Stochastic watershed—an analysis

B. Selig and C. L. Luengo Hendriks
in *Proceedings of SSBA 2012*, Swedish Society for Automated Image Analysis, pp. 82–85, 2012.

Abstract

Stochastic watershed is a novel segmentation method based on a probability density function. This probability density function is created by repeated realizations of seeded watershed with a fixed number of random seed points. We study the relationship between the algorithm's parameters and attributes of the image in order to find the strengths and the weaknesses of the algorithm. The stochastic watershed works best when the regions are of similar size. Additionally, the number of iterations can be minimized by choosing a number of markers close to the number of expected regions in the image. In the case where the regions have different sizes, the original algorithm yields unsatisfactory results.

Paper IV

Exact evaluation of stochastic watersheds: from trees to general graphs

F. Malmberg, B. Selig and C. Luengo Hendriks
in *Discrete Geometry for Computer Imagery*, ser. Lecture Notes in Computer Science, E. Barcucci, A. Frosini and S. Rinaldi, Eds., vol. 8668, Springer International Publishing, pp. 309–319, 2014.

Abstract

The stochastic watershed is a method for identifying salient contours in an image, with applications to image segmentation. The method computes a probability density function (PDF), assigning to each piece of contour in the image the probability to appear as a segmentation boundary in seeded watershed segmentation with randomly selected seedpoints. Contours that appear with high probability are assumed to be more important. This paper concerns an efficient method for computing the stochastic watershed PDF exactly, without performing any actual seeded watershed computations. A method for exact evaluation of stochastic watersheds was proposed by Meyer and Stawiaski (2010). Their method does not operate directly on the image, but on a compact tree representation where each edge in the tree corresponds to a watershed partition of the image elements. The output of the exact evaluation algorithm is thus a PDF defined over the edges of the tree. While the compact tree representation is useful in its own right, it is in many cases desirable to convert the results from this abstract representation back to the image, e.g, for further processing. Here, we present an efficient linear time algorithm for performing this conversion.

Paper V

Improving the stochastic watershed

K. B. Bernander, K. Gustavsson, B. Selig, I.-M. Sintorn and C. L. Luengo Hendriks

Pattern Recognition Letters, vol. 34, no. 9, pp. 993–1000, 2013.

Abstract

The stochastic watershed is an unsupervised segmentation tool recently proposed by Angulo and Jeulin. By repeated application of the seeded watershed with randomly placed markers, a probability density function for object boundaries is created. In a second step, the algorithm then generates a meaningful segmentation of the image using this probability density function. The method performs best when the image contains regions of similar size, since it tends to break up larger regions and merge smaller ones. We propose two simple modifications that greatly improve the properties of the stochastic watershed: (1) add noise to the input image at every iteration, and (2) distribute the markers using a randomly placed grid. The noise strength is a new parameter to be set, but the output of the algorithm is not very sensitive to this value. In return, the output becomes less sensitive to the two parameters of the standard algorithm. The improved algorithm does not break up larger regions, effectively making the algorithm useful for a larger class of segmentation problems.

Paper VI

The fast evalution of the robust stochstic watershed

B. Selig, F. Malmberg and C. L. Luengo Hendriks
submitted for publication in a conference proceeding, 2015.

Abstract

The stochastic watershed is a segmentation algorithm that estimates the importance of each boundary by repeatedly segmenting the image using a watershed with randomly placed seeds. Recently, this algorithm was further developed in two directions: (1) The exact evaluation algorithm efficiently produces the result of the stochastic watershed with an infinite number of repetitions. This algorithm computes the probability for each boundary to be found by a watershed with random seeds, making the result deterministic and much faster. (2) The robust stochastic watershed improves the usefulness of the segmentation result by avoiding false edges in large regions of uniform intensity. This algorithm simply adds noise to the input image for each repetition of the watershed with random seeds. In this paper, we combine these two algorithms into a method that produces a segmentation result comparable to the robust stochastic watershed, with a considerably reduced computation time. We propose to run the exact evaluation algorithm three times, with noise added to the input image, to produce three different estimates of probabilities for the edges. We combine these three estimates with the geometric mean. In a relatively simple segmentation problem, F-measures averaged over the results on 46 images were identical to those of the robust stochastic watershed, but the computation times were an order of magnitude shorter.

Paper VII

Fully automatic evaluation of the corneal endothelium from in vivo confocal microscopy

B. Selig, K. A. Vermeer, B. Rieger, T. Hillenaar and C. L. Luengo Hendriks
submitted for journal publication, 2015.

Abstract

Background: Manual and semi-automatic analyses of images, acquired in vivo by confocal microscopy, are often used to determine the quality of corneal endothelium in the human eye. These procedures are highly time consuming. Here, we present two fully automatic methods to analyze and quantify corneal endothelium imaged by in vivo white light slit-scanning confocal microscopy.
Methods: In the first approach, endothelial cell density is estimated with the help of spatial frequency analysis. We evaluate published methods, and propose a new, parameter-free method. In the second approach, based on the stochastic watershed, cells are automatically segmented and the result is used to estimate cell density, polymegathism (cell size variability) and pleomorphism (cell shape variation). We show how to determine optimal values for the three parameters of this algorithm, and compare its results to a semi-automatic delineation by a trained observer.
Results: The frequency analysis method proposed here is more precise than any published method. The segmentation method outperforms the fully automatic method in the NAVIS software (Nidek Technologies Srl, Padova, Italy), which significantly overestimates the number of cells for cell densities below approximately 1200 mm^{-2}, as well as previously published methods.
Conclusions: The methods presented here provide a significant improvement over the state of the art, and make in vivo, automated assessment of corneal endothelium more accessible. The segmentation method proposed paves the way to many possible new morphometric parameters, which can quickly and precisely be determined from the segmented image.