

UNIVERSITY OF TARTU
Faculty of Science and Technology
Institute of Computer Science
Data Science Curriculum

Marilin Moor

"FiBar": a Tool for Automated Analysis of Complex Biomaterials from Microscopy Images

Master's Thesis (15 ECTS)

Supervisor(s): Dmytro Fishman, PhD,
Marta Putrinš, PhD,
Karin Kogermann, PhD

Tartu 2023

"FiBar": a Tool for Automated Analysis of Complex Biomaterials from Microscopy Images

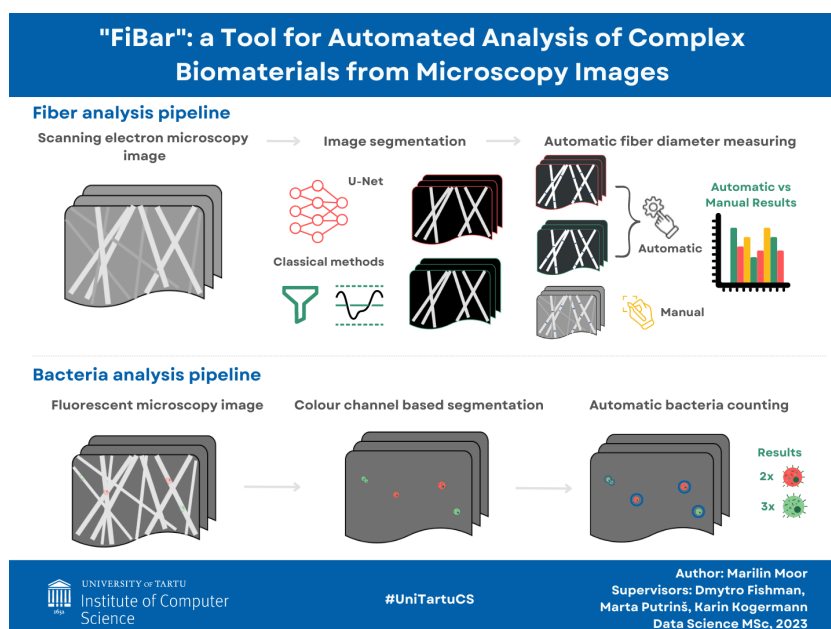
Abstract: The success or failure of many microbiological experiments depends on the image analysis of microscopy images, be it determining the livelihood of bacteria by measuring the fluorescence of individual cells or evaluating the quality of a fibrous mat by assessing the distribution of individual fiber diameters. Often a lot of image data is being generated from experiments, leading to a heightened demand of automated image analysis tools. This also holds true in the creation of complex biomaterials, which contain both fibrous textures and some other biocompound, like bacteria. Additionally, manual image analysis is deemed to be time inefficient and biased — both issues which this work aims to alleviate. This work presents the first version of "FiBar": a tool for the automated analysis of complex biomaterials. The tool consists of a fiber diameter measuring and bacteria analysis pipeline. "FiBar" was validated against other tools as well as manual measurements taken from microscopy images. The tool showed to be useful for speeding up the analysis while being relatively accurate.

Keywords:

Bacterial analysis, Electrospinning, Fiber analysis, Image analysis, Neural networks

CERCS: B230 Microbiology, bacteriology, virology, mycology; P176 Artificial intelligence; T111 Imaging, image processing; T490 Biotechnology

Graphical abstract:



"FiBar": komplekssete biomaterjalide mikroskoopia piltide automaatse analüüsi tööriist

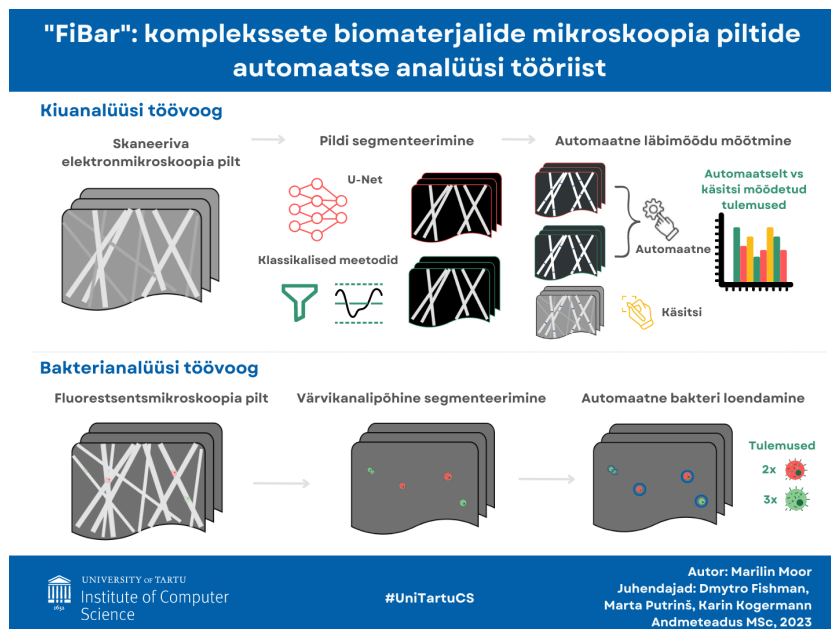
Lühikokkuvõte: Mikrobioloogia katse edu või ebaõnnestumine sõltub mikroskoopia pildi analüüsist, olgu see bakteri elumuse hindamine fluorestsentsi järgi või fiibermati kvaliteedi määramine kiudude läbimõõtude jaotuse põhjal. Tihti tekib katse käigus palju mikroskoopia pilte, ning see on suurendanud vajaduse automaatse pilditöötluse tööriista järele. See tõsiasi kehtib ka komplekssete biomaterjalide loomisel, mis sisaldavad nii kiulisi tekstuure kui ka muud biokomponenti, nagu bakterit. Lisaks sellele, et manuaalne pildianalüüs on ajakulukas on see sageli ka analüüsija poolt kallutatud - probleemid, mida selle töö lahendus püüab parandada. Töö tutvustab "FiBar"lahenduse esimest versiooni, mis võimaldab analüüsida komplekssete biomaterjalide mikroskoopia pilte. Lahendus koosneb kahest töövoost: kiudude läbimõõtude mõõtmine ja bakterirakkude analüüs. "FiBar"toimimist valideeriti nii muude programmide kui ka käsitsi mõõdetud tulemuste järgi. Lahendus kujunes kiireks ja võrdlemisi täpseks.

Võtmesõnad:

bakteri analüüs, elektroketrus, fiibri analüüs, pilditöötlus, tehisintellekt

CERCS: B230 Mikrobioloogia, bakterioloogia, viroloogia, mükoloogia; P176 Tehisintellekt; T111 Pilditehnika; T490 Biotehnoloogia

Visuaalne kokkuvõte:



Contents

1	Acronyms and abbreviations	6
2	Introduction	7
3	Background	9
3.1	Complex biomaterials	9
3.1.1	Imaging techniques for biomaterials	11
3.2	Biological image analysis software ecosystem	13
3.2.1	Assessment of fiber morphology	13
3.2.2	Estimation of the viability of bacterial cells	15
4	Methods and materials	18
4.1	Fiber diameter measuring pipeline	18
4.1.1	Image segmentation	19
4.1.2	Measuring diameters	23
4.1.3	Visualization of the results	26
4.2	Bacterial cell analysis pipeline	27
4.2.1	CZI conversion to image files	27
4.2.2	Bacteria counting using image processing methods	28
4.3	"FiBar" orchestration, data collection	29
4.3.1	Constraints of "FiBar"	31
4.4	Validation of pipelines	31
4.4.1	Fiber diameter measuring pipeline	31
4.4.2	Bacterial cell analysis pipeline	33
5	Results and discussion	36
5.1	Fiber diameter measuring pipeline performance	36
5.1.1	Horizontal scale bar test	36
5.1.2	Testing the diameter measuring algorithm	36
5.1.3	SEM image measurement tests	41
5.2	Validation of the bacterial cell analysis pipeline	51
5.2.1	Simulated green and red bacteria	51
5.2.2	Case study for bacterial cell analysis	52
6	Conclusions	56
7	Acknowledgements	57
	References	63

Appendix	64
I. U-Net architecture	64
II. Link to the project repository	65
III. Licence	66

1 Acronyms and abbreviations

Acronym/abbreviation	Full term
CZI	Carl Zeiss Imaging
DFOI	Distribution-Free Overlapping Index
DM	Diameter
ES	Electrospinning
GUI	Graphical User Interface
LIF	Leica Image File
OCR	Optical Character Recognition
OME	Open Microscopy Environment
OS	Operating System
PEO	Poly Ethylene Oxide
PLC	Copolymer of L-lactide and ϵ -Caprolactone
ReLU	Rectified Linear Unit
ROI	Region Of Interest
SEM	Scanning Electron Microscopy
TIF/TIFF	Tagged Image File Format
XML	Extensible Markup Language

2 Introduction

Visual inspection, like fluorescence microscopy has become an important tool in studying microbiology as well as pharmaceutical biotechnology [Xie et al., 2008]. This minimally invasive approach allows to observe bacteria in a controlled environment. Measuring the bacterial cell size and fluorescence intensity are just some features the scientist can gather from microscopy images. However, for research to be useful, the scientist has to generate a multitude of images. With the state-of-art microscopes, images can be acquired in large numbers and high quality, leading to bigger amounts of data to be analyzed.

For a higher throughput analysis, ample of methods for automatic and semi-automatic image analysis have been developed in recent years ranging in complexity - from simple image thresholding to deep neural networks [Eliceiri et al., 2012]. These methods have been applied in the software of multiple imaging instruments (Image Analysis Wizard Zeiss Microscopy), and free open-source solutions (e.g. DiameterJ/MicrobeJ plugin for ImageJ [Ducret et al., 2016, Hotaling et al., 2015], CellProfiler [Bray et al., 2015], ilastik [Berg et al., 2019]). Next to automation and speeding up the analysis of images, the tools provide means to measure quantities that cannot be easily assessed by the human eye. Considering the complexity of the microscopy images of biomaterials, no single out-of-the-box image analysis solution fits. Hence multiple aforementioned image analysis tools serve as inspiration in this work.

The general goal of the thesis was to design and develop an image processing tool - "FiBar" that can be used to analyze microscopy images of complex biomaterials within the Pharmaceutical R&D Lab at the University of Tartu. The tool consisted of two pipelines - one for measuring fiber diameters of the complex biomaterials, the other for the analysis of bacterial cells found in the sample. The performance of "FiBar" was tested on synthetic data as well as actual microscopy images. Additionally, a case study had been carried out by some of the supervisors - Marta Putrinš and Karin Kogermann - that was solved by utilizing the bacterial cell pipeline.

This thesis is done in collaboration with the Pharmaceutical R&D Lab at the Institute of Pharmacy which focuses on producing complex biomaterials, as well as the Biomedical Computer Vision Lab from the Institute of Computer Science at the University of Tartu. The work is funded by the Estonian Research Council funded research project no PRG 1507 (PI K. Kogermann): "Development of biorelevant assays for the analyses of multifunctional antimicrobial wound dressings for the treatment of wound infections".

The first part of the work provides an overview of what complex biomaterials are, additionally a selection of biological image analysis tools are introduced. The second part focuses on the image processing methods used in this work as well as the setup of the solution. The final part of this thesis covers evaluation of the tool's pipelines and provides suggestions on how it could be improved further. The architecture of the neural network used in this work, the link to the project code repository as well as the licence is provided in the Appendix. This thesis assumes prior knowledge of image processing

methods. In case of confusing field-specific constructs, the reader is suggested to refer to the Digital Image Processing textbook by Gonzalez and Woods [Rafael C. Gonzalez, 2018].

3 Background

This section aims to provide sufficient background for both the biological data used in this thesis and available software for biological image analysis.

3.1 Complex biomaterials

As quoted from the US National Institute of Biomedical Imaging and Bioengineering's (NIBIB) Glossary of Terms - "A biomaterial is said to be any matter that interacts with biological systems" [National Institute of Biomedical Imaging and Bioengineering, 2023]. Conventionally, biomaterials were considered to be non-living materials implemented in medical devices (e.g., joint replacement, contact lenses) [Tathe et al., 2010]. Due to the recent advancements in the field, some biomaterials are now utilized as wound dressings where controlled cell-to-cell and cell-to-material interactions can occur [Naderi et al., 2011]. This in turn, serves as a suitable medium for quantitative cell adhesion and growth which is of utmost importance in wound recovery [Davis et al., 2005].

One method to produce wound dressings is electrospinning (ES). ES produces fiber mats from polymers (such as starch, and gelatine) with fiber diameters of nano- to micrometer range and with controlled surface morphology. This technology has proven to be especially compatible with wound healing as the structure of the fiber mat can mimic the skin's extracellular matrix, which is pivotal for supporting cell growth [Haik et al., 2017] (refer to **Figure 1** for comparison of the fiber mat to natural tissue).

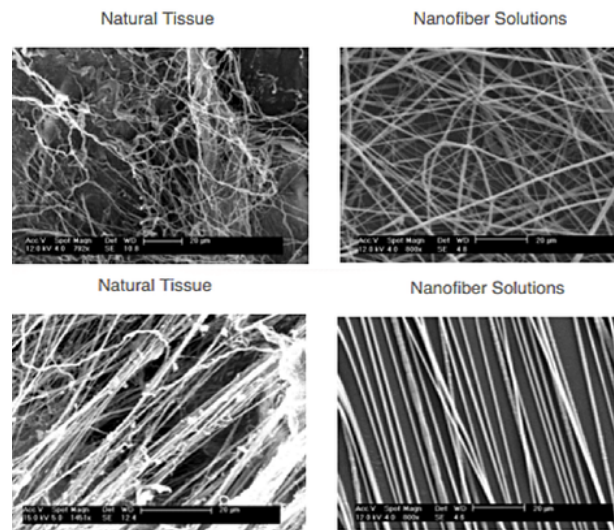


Figure 1. Examples of fibers in natural tissue compared to fibers in the electrospun mats [Matrix, 2023].

As visualized in **Figure 2**, the electrospinning technique can vary in multiple parameters: the polymer used for creating the material, humidity in the ES chamber, and the voltage used to activate the electrospinning.

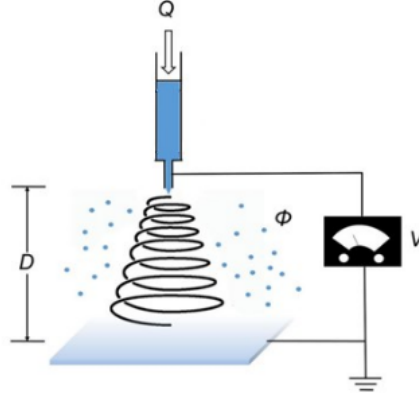


Figure 2. Representation of the experimental setup and adjustable parameters of (monoaxial) electrospinning. The polymer solution is in blue. Q: flow rate, V: high voltage, D: distance from the syringe nozzle to the collector, ϕ : relative humidity (modified [Weng and Xie, 2015])

The key points for describing the morphology of the electrospun (ES) wound dressings are fiber size, pore size, and surface topography features [Lanno et al., 2020]. It has been addressed that the fibers' structural properties affect the wound-healing process. For instance, the high porosity of the fiber allows oxygen and water permeability, nutrient exchange, and removal of metabolic waste [Kamble et al., 2017]. Additionally, research has highlighted that the fiber morphology features are connected to the mechanical properties, for instance, due to the fiber chain orientation and fibers' structural order, fibers with smaller diameters tend to be stronger [Yao et al., 2014].

Unsurprisingly, the morphology of ES fibers in the fiber mat depend on the ES process and the parameters set for ES - such as the voltage, polymer solution etc. [Preem and Kogermann, 2018]. Thus, there is demand for fiber characterization tools that enable to estimate the fiber properties accurately, thereby providing insights into how successful the ES was.

The heterogeneous structure of the biomaterial is what makes the material complex: it may consist of multiple polymers or even contains a polymer and some living matter or other compound [Zusman, 2010]. Scientists have proved that it is possible to encapsulate different biocompounds within the fiber, such as medications [Lanno et al., 2020, Meinel et al., 2001], and bacteria [Klein et al., 2009], that can ultimately boost wound healing. More specifically, the encapsulation of bacteria has been shown to improve the stability of cell enzymatic activities and protection from environmental stresses. This is very

beneficial when creating a wound dressing where the bacteria have been modified to release wound healing compounds [Klein et al., 2009]. With this in mind, the interaction between the bacteria and the material also adds to the complexity. Alongside, if not all, then most species, the bacteria also necessitate to live in an environment supporting their growth and metabolism for them to function properly. Research has shown that the ES method provides a suitable enough environment for bacteria. For example, in the works of Salalha et al., the *Staphylococcus albus* remains completely viable while some *Escherichia coli* cells lost their colony-forming ability after they were ES into the polymer. Additionally, all the bacterial cells retained their viability 3 months after ES [Salalha et al., 2006]. It should be pointed out that the ES material with bacteria applicable for wound healing is one of the main biomaterial of interest throughout this work.

3.1.1 Imaging techniques for biomaterials

To assess the mechanical and biological properties of the ES fibers, multiple imaging methods can be employed. These methods roughly branch into two based on the target:

- evaluation of the fibers' properties;
- estimating the stability and viability of living cells encapsulated in the fiber.

Both of the aforementioned goals can be achieved by using various microscopic techniques, such as scanning and confocal microscopy.

Scanning Electron Microscopy. Scanning Electron Microscopy (SEM) is a technique that can provide information about the sample's topography, chemical composition, and electrical behavior of the upper layer (1 μm depth). The basic working principle can be concluded as follows: the setup has an electron gun that blasts out electrons (with energies from 2-40 keV), the interaction between the electron beam and the specimen (which is laid on electrically conducting pads) is collected by various detectors, which signal values are projected onto a micrograph (an example is provided in **Figure 3**). Depending on the microscope, different magnifications can be acquired - ranging 10x to 500000x (from this point onward the "x" is being omitted when referring to magnification) [Vernon-Parry, 2000].

Due to the size range of the diameters of ES fibers, this technique is often utilized to evaluate nanofiber mats [Lanno et al., 2020, D'Amore et al., 2010]. The micrographs enable the interpretation of the structure of the fiber - porosity, elongation, uniformity - both qualitatively and quantitatively. For the latter approach, the micrographs are exported as image files. The selection of the image format may vary depending on the software that is provided with the microscope. These SEM images are then used for measuring the diameters of the fibers, as well as the porosity and orientation of the

surface fibers. Furthermore, attempts have been performed to also understand the fiber mat porosity in depth using specific calculations and predicting the 3D pore volumes [Liu and Hwang, 2012]. The approach to automatically measure these features is discussed in the section "Assessment of fiber morphology".

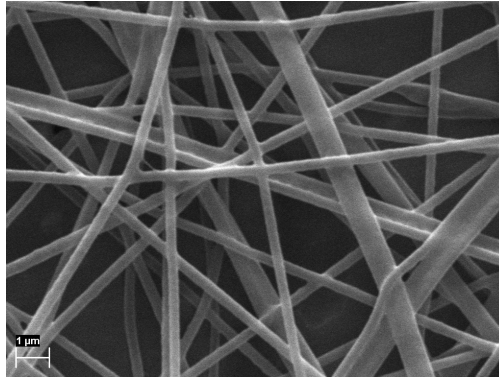


Figure 3. Example of a SEM image.

Confocal Microscopy. Confocal microscopy is an imaging tool that uses fluorescent light to display a section of the sample. To create an image of the sample, the focal spot of the microscopy objective is scanned in the X-Y plane, and the sharp, high-contrasted image is constructed one pixel at a time (some microscopes can handle multiple points at once) [Zeiss, 2023]. Although the sample is not actually sectioned, it is possible to acquire a stack of confocal image planes. For fluorescent light microscopy, biological dyes (fluorophores) need to be added to the specimen for it to react to light at a compatible wavelength. These fluorophores greatly improve image quality by targeting only specific parts of the sample and as a result decreasing the overall signal-to-noise ratio [Nwaneshiudu et al., 2012].

As confocal microscopy enables localization of specific areas of the sample, it is ideal for assessing the viability of encapsulated cells in the fiber. This, of course, necessitates the cells to be modified so that they emit fluorescent light when excited [Zupančič et al., 2019]. In the case of bacterial cells, fluorescence expression does not directly link to bacterial activity, therefore special nucleic acid coloring or fluorescent protein plasmids encoding for the expression of fluorescent proteins are utilized. These options enable monitoring the viability via expression of a specific fluorescence. For instance, the LIVE/DEAD BacLight Bacterial Viability Kit by ThermoFisher stains cells that are considered to be dead or dying in red, whereas cells with an intact membrane will be stained green [ThermoFisher Scientific, 2023]. An example of green and red stained bacteria in a sample is provided in **Figure 4**.

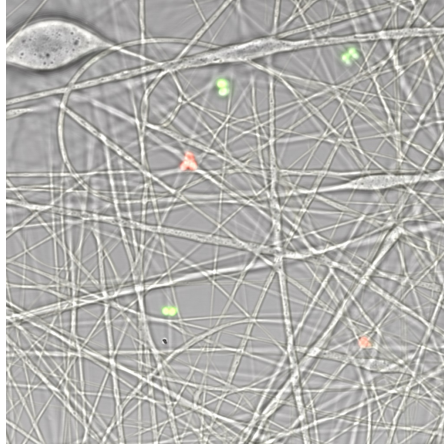


Figure 4. Example of a microscopy image with green and red stained bacteria and fibers.

3.2 Biological image analysis software ecosystem

This subsection focuses on the efforts made thus far for the analysis of biological images at a fiber and encapsulated cell level.

3.2.1 Assessment of fiber morphology

Commonly, the gold standard for evaluating nanofiber morphology has been manual measurements in SEM images using a graphical tool in image analysis programs, such as ImageJ [Collins, 2007]. However, this approach is time-consuming and low throughput. Additionally, selection bias is very likely [Begg and Berlin, 1988], as experimenters may omit fibers that are non-representative - such as very thin or thick fibers. As the distribution of fiber diameters is one of the key metrics to describing the fiber morphology [Lanno et al., 2020], a sufficiently large number of fibers needs to be measured manually to reduce the standard deviation to a level where meaningful statistical comparisons can be undertaken.

To speed up the process of collecting manual measurements and decrease selection bias, a variety of tools have been developed to assess properties, such as the nanofiber orientation [Rezakhaniha et al., 2011], fiber length [Giusti et al., 2018], hole size [Hotaling et al., 2015], and nanofiber diameters [Tomba et al., 2010]. As automating fiber diameter measurements is one of the key focus points in this thesis, the most relevant works in this regard will be discussed in further sections.

The fiber diameter measurement process can be divided into the following parts: image processing to extract fibers, measuring fiber diameters from the segmented image, and validating the results.

Image segmentation to extract the fibers. Most papers have approached fiber extraction using classical or more advanced image processing methods, which are applied to the whole image at once [Hotaling et al., 2015] or sub-images of the original image [Tomba et al., 2010], [Chen et al., 2010].

The authors of some papers have been able to successfully extract fibers from the image by employing only classical image processing methods. In the paper by Chen et al., noise was filtered with kernel convolution, and Otsu thresholding was used to binarize the resulting image. The image and fiber edges were detected using Sobel gradients [Chen et al., 2010]. In the works of Tomba et al., nonlinear histogram equalization was employed for enhancing the pixel contrast, which was followed by denoising using a wavelet filter. For fiber segmentation, multiple methods were applied, such as multivariate image analysis and straight-line approximation [Tomba et al., 2010].

DiameterJ is a diameter measuring tool deemed to be the front runner in this field, as it provides different options to segment the fibers from the image. This is possible because it is part of the ImageJ ecosystem, where other segmentation algorithms/plugins can be accessed [Hotaling et al., 2015]. The thesis author’s testing revealed that the DiameterJ’s most fruitful method to segment fibers was statistical region merging.

An example of deep learning-based segmentation can be found in a web application named FiberDiameter which was released in November 2022. The solution runs a pre-trained U-Net model in the back-end that segments fibers from the uploaded image. The model had been trained on synthetic as well as real-life images provided by the community [Fernando Cossio, 2023].

Measuring fiber diameters from the segmented image. The second part of the pipeline, namely measuring fiber diameters, has been approached differently in the literature. For example, in the efforts by Tomba et al, the orthogonal distance between the segments, which represents the right and left edge of a fiber, is estimated [Tomba et al., 2010]. In the work by Chen et al., the shortest path algorithm is adopted. For this, a random point is chosen at the left edge (1 fixed point), another point is chosen from the right edge (10 points chosen altogether from the opposite edge), and the distance between the points is calculated. The shortest distance out of the 10 measurements is set as the diameter of the fiber. Despite stating that the diameters can be measured from up to down, the authors do not describe how this was achieved [Chen et al., 2010].

DiameterJ provides two options to assess the diameter of the fiber, either via the superpixel diameter or using a histogram from multiple diameter measurements. The former approach succeeds by dividing the areas of all the fibers (foreground white pixels are summed up) by the total length of all the fibers’ centerlines. The centerlines are obtained through thinning or the Voronoi spatial tessellation (connecting the background black pixels). This technique assumes that fibers are long and rectangular. The second approach for fiber diameter measurements using DiameterJ is performed by transforming




the segmented image with the Euclidean distance transform algorithm. Thereafter, the Voronoi centerlines are overlaid with the transformed image. Intersections between the centerlines and values in the distance-transformed image are omitted. The grayscale values under the leftover lines (ideally located in the middle of the fiber) are obtained and multiplied by 2 [Hotaling et al., 2015]. While the distance transformation step was adopted in the solution of this work, the diameter measuring part of the DiameterJ (version 1.018) tool did not work at the time of testing.
















Validation of the results. The distribution of measured fiber diameters is often visualized as a histogram. Some statistical metrics, such as the mean and standard deviation, and coefficient of variation, are also used to describe the distribution. Generally, manual measurements are used as a reference which the automatically measured results are compared to [Tomba et al., 2010, D’Amore et al., 2010].

3.2.2 Estimation of the viability of bacterial cells

To determine the viability of bacterial cells, the cells ought to be detected first. This is generally achieved by thresholding. However, the structure of bacterial cells is known to vary between different cell types (rods, coccus etc.) or even within the lifespan of one bacterium. This directly affects the choice of parameter sizes in the segmentation process and is one of the reasons, why multiple analysis tools have been developed. **Table 1** provides a curated overview of the current state-of-art tools for bacterial image analysis.

Firstly, it should be pointed out that many file formats exist when it comes to microscopy images. One way to explain this is the ample amount of microscopy devices on the market that have their own method for building images. Some examples of these formats are also provided in the **Table 1** - LIF belongs to the Leica Microscopes ecosystem, whereas, the CZI format has been created by the Carl Zeiss Microscopy enterprise and is most compatible with the company’s own ZEN imaging software [Collins, 2007], [GmbH, 2011]. The disparity between formats urged the Open Microscopy Environment (OME), to build and release Bio-Formats. This is a Java-based file conversion tool used across many platforms (like CellProfiler as is evident in the aforementioned table) [Moore et al., 2015]. Additionally, an OME-tiff format can be viewed in the BactMAP and Oufiti software column. This is a standardized format established by the aforementioned OME consortium, that is essentially a multi-page TIF image. Next to the developed image analysis software tools, the OME-TIFF format is supported by multiple microscopy company softwares, such as the preceding Leica Inc. and Carl Zeiss Microscopy, as well as PerkinElmer and Scientifica [Environment, 2018].

Table 1. Comparison of features in different microscopy image analysis tools for bacteria.
(   represent the Linux, Mac, Windows operation systems respectively)

	CellProfiler	BacStalk	BactMAP	Oufti	BiofilmQ
Suggested use case	Different eukaryotic and prokaryotic cells	Common and stalk-forming bacterial cells	Fluorescent bacterial cells, integrating bacterial data from other programs	Touching bacterial cells of varied morphology	Microbial biofilm communities
Supported input file format(s)	Most imaging formats from Bio-Formats	JPEG or TIF	OME-tiff	OME-tiff	Few formats from Bio-Formats, e.g. LIF, CZI, TIF
Outputs	Collected data table	Collected data table, varied plotting options	Different visuals for fluorescent cells	Collected data table and visualizations	Collected data table and visualizations
GUI	✓	✓	✗	✓	✓
Development language	Python	MATLAB	R	MATLAB	MATLAB
Download requirements	 /  (from source: )	 /  /  & MATLAB R2014b+	 /  /  & R	 /  (from source:  with MATLAB)	 /  /  and MATLAB R2017b+
Paper reference	[McQuin et al., 2018]	[Hartmann et al., 2020]	[van Raaphorst et al., 2020]	[Pai, 2016]	[Hartmann et al., 2021]

The software programs in **Table 1** are stand-alone tools, often purposed for specific cases and complex to use as separate modules. Alas, not all of the tools suit the data provided in this work or are too complicated to solve the issue at hand.

As one of the goals in this work is to create a pipeline for microscopy images with bacterial cells, endeavours by Chiang et al. and the ImageJ's module "Analyze particles" have served as a source of inspiration [Chiang et al., 2015, ImageJ, 2023]. In the work of Chiang et al., the goal was to count colonies from an image of a Petri dish. One of the efforts of this work went into extracting the dish from the image, so it would not be interfering with the colony counting. The extraction of the colonies, however, was achieved through classical image processing methods, such as the bottom-hat transform, erosion, Otsu's thresholding and watershedding. For validation of their approach, they compared their system to two other colony counting systems - NIST's Integrated Colony Enumerator (NICE) and Clono-Counter. After analyzing 21 images of colonies on Petri dishes, the proposed approach surpassed both counting systems based on the average of precision, F-measure and recall [Chiang et al., 2015].

Next to having an ImageJ plugin for fiber measurements (DiameterJ), a plugin for bacterial cell detection and analysis is also existing - MicrobeJ. By using conventional thresholding techniques, MicrobeJ can detect bacterium of any morphology and generate subpixel resolution contours, additionally the medial axis is being extracted to determine the geometrical and topological properties of the cell shape [Ducret et al., 2016]. As this tool is focused more on single-cell analysis, it is also limited to having input images of very high resolution. For this reason, ImageJ provides a module named "Analyze Particles" for automatic particle counting. This takes a segmented image as an input with no overlaying objects in the image. Both of these conditions can be met using the built-in modules (Auto-thresholding, Watershed operation) of ImageJ. The particle analysis plugin allows the user to filter out particles based on size, as well as circularity. Finally the collected results are output as a data possible, it is also possible to obtain an image with detected bacteria [ImageJ, 2023].

4 Methods and materials

This part of the work discusses the methods used in the two pipelines - fiber diameter measuring pipeline and bacterial cell analysis pipeline. Additionally, the techniques and data used to validate the pipelines' performance will be elaborated later. This includes details on how the data was created as well as what metrics were considered for testing.

4.1 Fiber diameter measuring pipeline

The overview of the complete proposed fiber diameter pipeline is provided in **Figure 5**. As discussed prior, SEM is a method for analyzing the diameter range of the ES fibers as well as ES fiber mat morphology. Hence the input of this pipeline was an image of a SEM micrograph. This image was then segmented into fore- and background via two separate approaches - by using classical image processing methods or by utilizing a pre-trained U-Net model. The extracted fibers in the upper layers of the SEM image belonged to the foreground, all other parts in the image were regarded as the background. Later, the fiber diameter measurements were obtained using a custom algorithm that took the segmented image as input. The gathered diameters were then converted to nanometer scale by fetching the value and scale from the source image, and a corresponding histogram of the measured diameters was produced. This histogram was then used for the evaluation of the pipeline. The aforementioned parts will be expanded in further detail below.

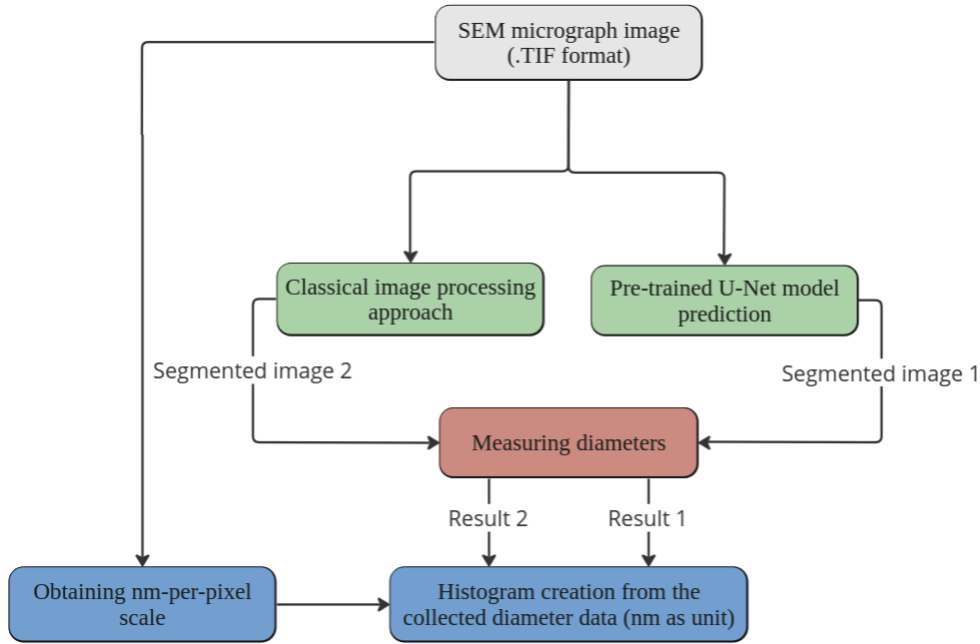


Figure 5. General workflow of the fiber diameter measuring pipeline. The flow was initiated with an image of a SEM micrograph (gray box), which was segmented either classically or using a pre-trained U-Net model (boxes in green). Diameters (initial unit: pixel) were measured from the segmented images (red box) and were then converted to nanometers based on the scale provided and the histogram was created (blue boxes).

4.1.1 Image segmentation

Image segmentation in this application is the process where the uppermost fibers of the SEM micrograph are considered to be foreground class, and all the rest of the image is considered to be background class. As methods in this field have evolved significantly over the past ten years, two distinct approaches were tested for SEM image segmentation: classical and deep learning-based.

Classical image processing approach. In the classical image processing technique, the raw image was first converted into grayscale. Then, a Kuwahara filter (3x3) was applied to the image. Generally, filtering is applied to remove noise and distortion. The filtering process entails a moving window that scans over every pixel in the image, and based on the pixel values in the window, the center pixel will obtain a new value. The common operation for filtering is taking the mean or the median of the pixels in the window. However, this sort of smoothing may affect the parts of the image where edges

are important. Such was the case for these SEM images where the edges of the fibers marked the start points for diameter measuring, and thus, ideally, the fibers' size on the segmented image should not differ compared to the original image. The Kuwahara filter is one example of smoothing that does not affect the sharpness of contours [Bartyzel, 2015]. The Kuwahara filter converges from the previous filters as the filter window (with an odd-numbered edge size) itself is divided into 4 equal areas - meaning that each subarea will have overlapping pixels (refer to **Figure 6** for clarification). For each subsection, the average and variance are calculated. The resulting value of the center pixel is the average mean of the section with the smallest variance.

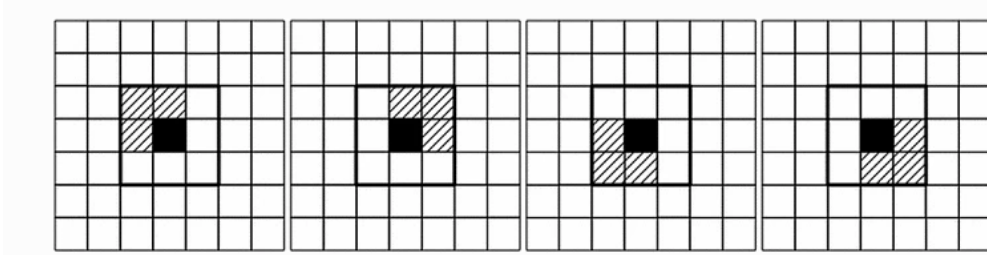


Figure 6. An example of a 3x3 Kuwahara filter window with 4 shaded subareas, the black pixel signifies the central pixel of the window [Bartyzel, 2015].

To extract the fibers from the background of the filtered image, Huang thresholding was undertaken. Compared to the infamous Otsu thresholding, Huang thresholding considers the fuzziness of the image. In short, provided a certain threshold value, the membership function of a pixel is defined by the absolute difference between the gray level and the average gray level of its belonging class (foreground or background). The larger the difference, the smaller the membership function becomes, thus minimizing the fuzziness of the image [Huang and Wang, 1995]. The output was a binary image where the foreground class was of white color and pixels belonging to the background class were black. In order to remove some white artifacts in the background and smooth out the image, a median filter (15x15 window) was applied. The intermediate results of a sample image are provided in the **Figure 7** below.

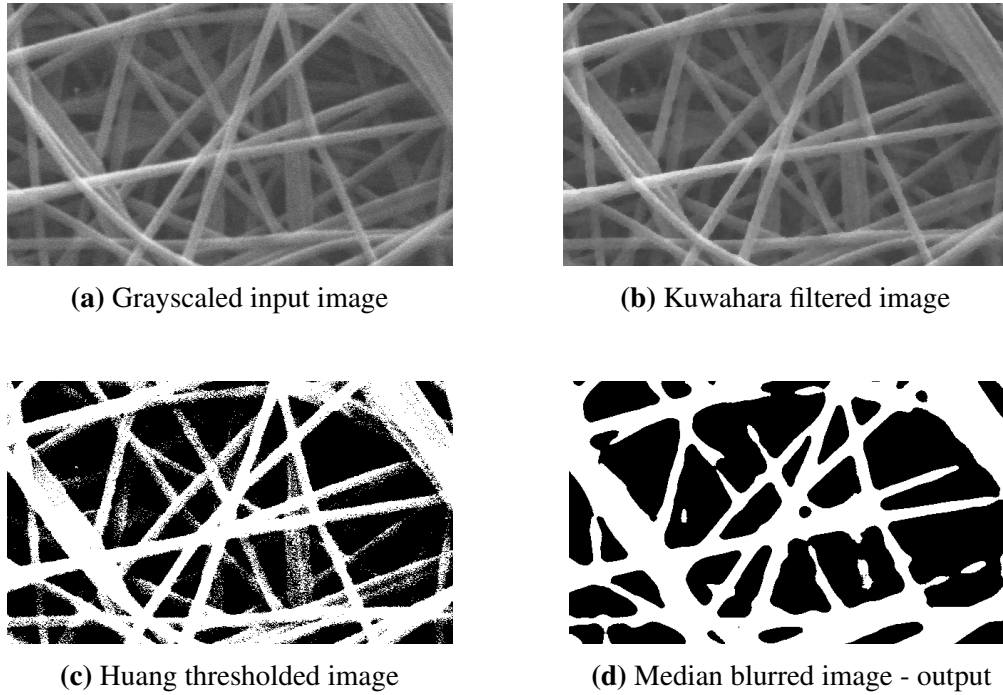


Figure 7. Intermediate results (from a to d) of the classical segmentation approach. The images are cropped by 200 pixels from all edges. While visually difficult to notice, the smoothed texture of the fiber in b is what differs from a.

Pre-trained U-Net model prediction. U-Net is a well-known network for biological image segmentation [Ronneberger et al., 2015]. The network architecture has a U-like shape, which consists of a contractive path and an expansive path. In the final layer, a 1x1 convolution is implemented and the output is sigmoidally activated to map each component feature vector to the desired number of classes (e.g. foreground and background class). The model architecture in this thesis mostly followed the original setup, the network architecture is provided in Figure A1. Convolutional layers were implemented with 3x3 sized kernels and the ReLu activation function. Dropout of 0.1 between the layers was used, which randomly silences 10% of all neurons in the hidden layer, and the learning process was guided by the Adam optimizer with a learning rate of 0.0002.

This network was trained by using one SEM image (with 15k magnification) and its corresponding segmentation map (the segmentation was manually done in Label Studio). The grayscale input image (with shape (768, 1024)) was cut into 100 256x256 overlapping sub-images. The same procedure was done with the segmented image. The model was trained on 90 sub-images and validated on the remaining 10 images. **Figure 8** shows the training process. Since, only one SEM image was manually segmented, there was no possibility to create an independent validation or test set for this model, instead, it

was attempted to overfit the data purposefully, relying on the assumption that other SEM images of fibers will be sufficiently similar to the image the model was overfitted on.

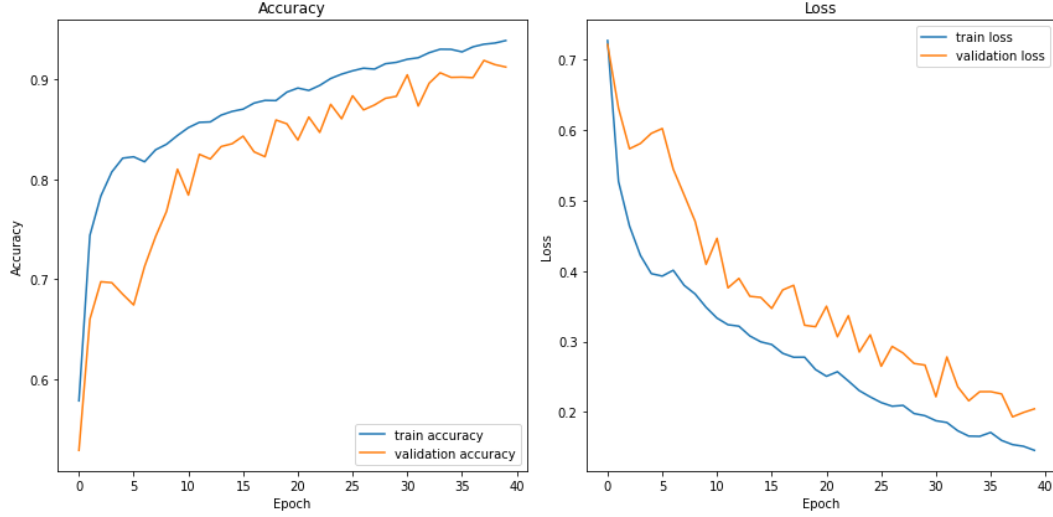


Figure 8. The learning curves of the U-Net model. The overfitting is not striking here as the images used for validation were from the set of subimages described above.

The model was then tested with an unknown SEM image (also 15k magnification). The test image was cut into unique patches of size 256x256. After setting these patches as input for the model, the predictions were stitched back together. The values in the output image were thresholded with model probabilities (ranging between 0 and 1). The pixel value in the output was set to white, if it was higher than a provided prediction probability, and black otherwise. The probability threshold ($p=0.6$) was fixed by overlapping the segmented image with a Canny edge detected input image which detects the edges of the foreground fibers relatively well (see **Figure 9** for comparison between probabilities of 0.6 and 0.9). This allowed to evaluate how well the model predicted the classes as well as to ensure that the dimensions of the fibers would not differ too much compared to the original.

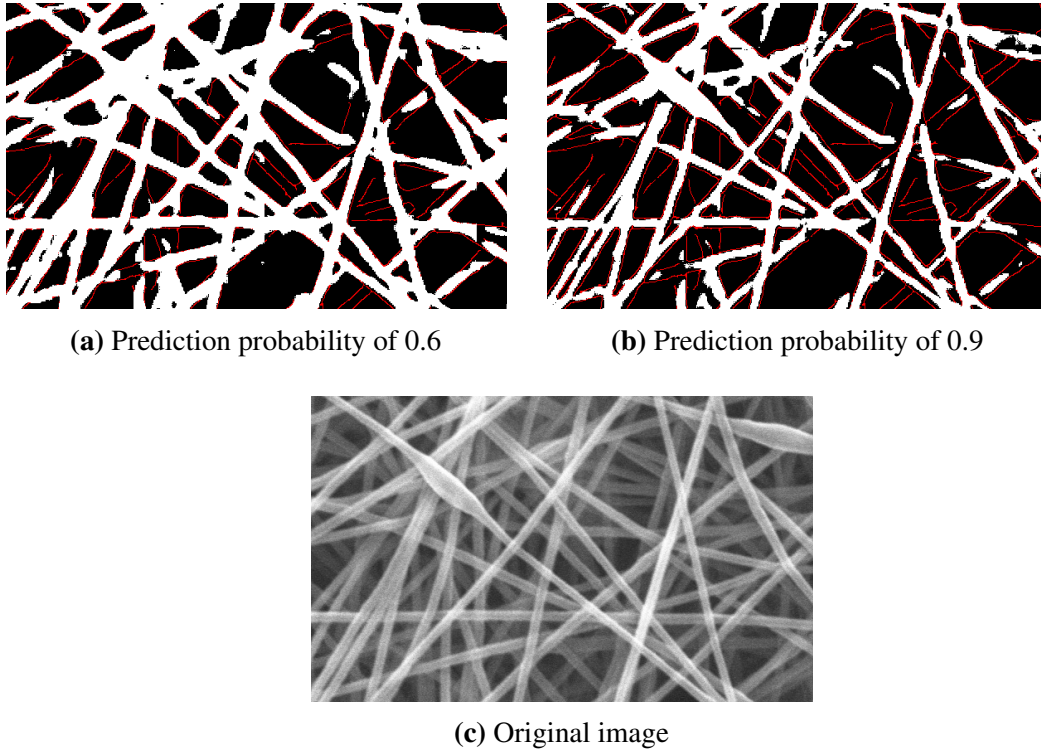


Figure 9. The output images thresholded with prediction probabilities of 0.6 (a) and 0.9 (b) with the original input image (c). The areas in black are the background class of the image, the sections in white belong to the foreground class and the pixels in red are the detected edges from the original image. The images are cropped by 200 pixels from every edge.

It can be seen from **Figure 9b** that segmentation with high prediction probability (0.9) reduced the amount of detail quite a lot and classified some foreground fibers (upper-right corner) as background. Hence it was opted for a smaller probability of 0.6.

Once the model's performance was qualitatively assessed, the model was saved as .h5 format and thus iteratively purposed for the segmentation of SEM images. Despite the predictions in the figure above not fully filling in the area between the edges, they had managed to do it partially, and was sufficient for the next step of diameter measuring. It should also be pointed out that the outputs often had a peculiar stair-like spiky artifact on the edges that were smoothed out with a 15x15 median filter.

4.1.2 Measuring diameters

The diameter measuring process can be sectioned into three: transformations of the image, including skeletonization, pruning and distance transform, starting point selection

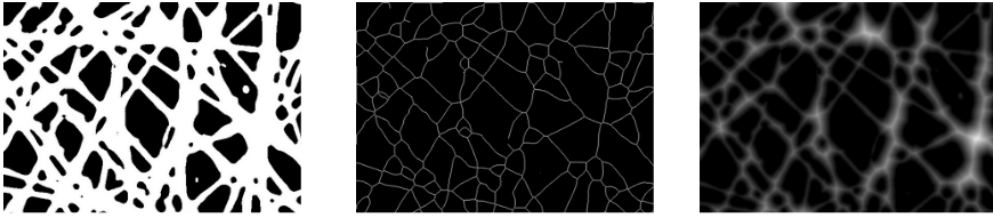
and applying the diameter measuring algorithm. The main steps of the process are gathered in **Figure 10**.

Skeletonization, pruning and distance transform. Before the diameters were to be obtained from the image, a skeletonized and pruned version of the segmented image was generated. Skeletonization (also known as "medial axis", "thinning") of the image allowed to find the points in the foreground class (the fiber) that are the furthest from the background. Consequently, the connectivity of the skeleton was not so conserved, as unwanted branches occurred, which would have interfered with the following steps. This issue was alleviated by pruning (hairs < 50 pixels removed), a method which many researchers, including Montero and Lang, have deemed sufficient [Montero and Lang, 2012]. It should be pointed out that pruning was not done for images of 2k and 5k magnification, as it was not capable of pruning suitable size elements within 10 minutes (Intel i5 8th gen processor). This is likely because of the larger density of information in the image that leads to the difficulty to differentiate between hairs and the parts of the skeleton. The distance transform of the image was also needed. As the distance transform finds the closest zero pixel for each pixel of the image, it allowed to determine the distance values along the medial axis in pixel units. For the distance transform, the Euclidean distance with a 3x3 mask was used as a distance measure [OpenCV, 2023].

Starting point selection. The starting points are locations on an arbitrary edge of the foreground class from which the diameter is measured (refer to **Figure 10b** for an example). The point locations were chosen randomly based on the dimensions of the image. However, the point location needed to fulfill two conditions: the point is at least one pixel away from all image edges and the two neighboring pixels should be black. The former condition avoided erroneous measurements taken too close to the border. At the same time, the latter increased robustness to noise - for example, a single white pixel in the background will not be a starting point. It also decreased erroneous measurements in obscure situations, such as an edge pixel being in the arc of V-shaped fibers. The number of unique starting point locations depends on the use case - generally, 100 diameter measurements per image are considered statistically valuable.

Diameter measuring algorithm. This paragraph covers what happens once the starting point has been established. The diameter measuring algorithm consisted of the following stages: direction finding, finding the nearest white pixel on the skeletonized image, and saving the diameter values. The stages are visualized in **Figure 10c**. Finding the direction of the fiber was estimated from the quarter (or quarters) with the most amount of white pixels. This was achieved by creating a 13x13 window (blue box in **Figure 10c**) where the starting point was the window's center point and the amount of white pixels in each quarter was collected. With the direction fixed, the nearest white pixel was to be attained.

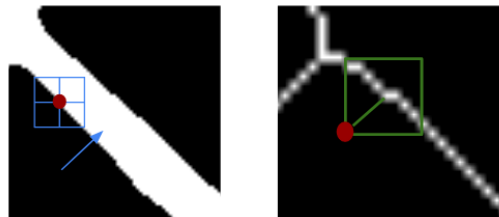
The first step was to determine the region of interest (ROI) which reached the medial axis from any starting point - the edge size of this bounding box was a rounded up maximum value of the distance transform. In cases where the starting point was closer to an edge than this aforementioned window edge value, the region's edge size was set based on the distance from the point to the closest edge. Based on the known direction and the starting point, the ROI was positioned so that most of the space under the ROI (green box in **Figure 10c**) would include white pixels. The location of the closest white pixel was used to find the distance value from the same coordinate in the distance transform. This distance value was then multiplied by two to obtain the diameter of the fiber in pixel units.



(a) Examples of the segmented, skeletonized and distance transformed image



(b) Starting point on the segmented image and the zoomed-in region of the starting point



(c) Direction finding, closest white pixel determining from the skeletonized image

Figure 10. Main steps taken to measure diameters - transformations, starting point assignment and closest white pixel determination. Here just one starting point is taken to show the process (b). It can be seen (c) that the direction is towards the upper-right quarter as this has the most white pixels. After the location of the closest white pixel is determined, the distance value is taken from the distance transform and multiplied by 2.

4.1.3 Visualization of the results

To comprehend the fiber diameter characteristics, the distribution of the measurements were visualized. In case of having more than one image of a sample, the measurements were combined into one distribution. In this thesis, the visualizations of distributions were beneficial to understand the inter-approach (automatic vs manual) differences. However, comparing values of these distributions assumed that they all were based on common units, which in this case was nanometer (nm). When at first it was only possible to estimate the diameter of fibers in pixels, while later, provided the magnification of the image, it was possible to convert pixels to nanometers. To enable this conversion, the scale (often at the bottom left) in the SEM image needed to be processed (see example of scales in **Figure 11**).

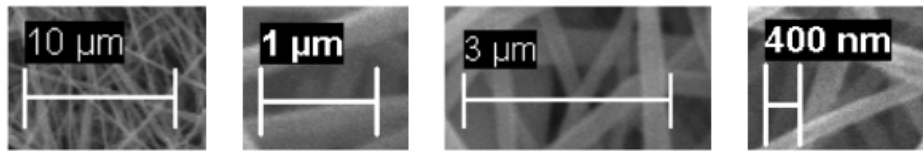


Figure 11. Examples of scales with varied values and units fetched from the SEM data provided.

The scale was a horizontal bar that shows how many pixels on the image correspond to a value-unit which was usually printed at the top of the scale. As the number of pixels per nm was never constant, the issue prompted two tasks to be solved: identifying the value and unit using optical character recognition (OCR) and obtaining the length of the horizontal bar.

The approach assumed the unit and value to be highlighted in black. This enabled to segment the corresponding area (with the value and unit) using simple binary inverse thresholding. After, contours were detected from the thresholded image. In this case, the biggest contour was always around the value-unit part of the image - this knowledge allowed to establish the exact location of the value-unit section in the image. This value-unit part was then provided as input for OCR. The Python-tesseract, a Python wrapper for Google's Tesseract-OCR Engine was utilized for this [pyt, 2023]. Initially, Pytesseract's OCR system was very erroneous when the value-unit compound was provided as input, so the value and unit were split. The splitting was achieved by determining the range of the biggest amount of consecutive black pixels in the middlemost row of the contoured value-unit region. The separated unit and value were then zero-padded from all sides. Thereafter, the unit and value were determined by setting Pytesseract's page segmentation mode (psm) to 10 which referred to treating the whole image as one character. The OCR process was cyclic where with every iteration the unit and value image dimensions were increased by 1 pixel (height and width ranged from 10 pixels to 30 pixels). The output

of the OCR was compared to a list of known values and units. If Pytesseract’s output matched with any of the reference values, the cycle broke.

The horizontal bar was assumed to lay below the contoured value-unit compound. Thus the region below the contour was cropped and thresholded, so that the bar (in white) was extracted from the image. The length of the bar was determined through the difference of the minimum and maximum column coordinate of the white pixel values. The amount of pixels-per-nanometer was found through the division of the value in nanometers (the μm was converted to nm) by the amount of pixels. This scalar was then used to convert all of the diameter values in pixels to nanometers and distributions of the measurements could be observed.

4.2 Bacterial cell analysis pipeline

The outline of the bacterial cell analysis pipeline is provided in **Figure 12**. The pipeline commenced with input data in CZI format which was converted to suitable color channels based on the file’s metadata provided in XML format. Then, a sequence of image processing methods was applied which resulted in the determining the amount of red and green bacteria within one sample. The count results would then be further analyzed by researchers.

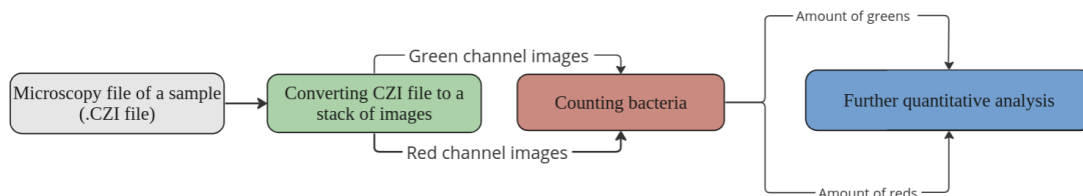


Figure 12. General workflow of the bacterial cell analysis pipeline. The flow was initiated with a microscopy CZI file. This file was converted to stack of images and the image arrays for the red and green channel were extracted. Images for both channels were then used for bacteria counting.

4.2.1 CZI conversion to image files

The CZI data storage schema consists of numerous segments, one of the most important being the metadata. Thanks to the Aicsimageio and Czifile library in Python it was possible to access the metadata of the CZI file [Gohlke, 2023, aic, 2020]. Some of the most useful features collected were the following: channel color names (to ensure the colors in the file were compatible with the script), amount of planes (images) in the z-axis in one CZI file. Provided the shape of every z-plane, it was possible to convert the

plane to a NumPy array, reshape and split the array based on the color channels provided in the metadata. After, the array of every color channel was converted to 8-bit and the images were ready to be processed for bacteria counting.

4.2.2 Bacteria counting using image processing methods

First, in order to keep the amount of pixels per bacterium in the same size range, all of the images were uniformly resized to shape 512x512. This enabled to use a constant kernel size for morphological operations. Most of the output results of the following intermediate steps are provided in **Figure 13**.

The grayscale image of a specific color channel was thresholded so that the value below a certain threshold (pivot intensity of 200) would turn black but otherwise the pixel value would remain the same (refer to **Figure 13b** for an example). This allowed to study the intensity changes within a cluster of bacteria: the more intense regions were used for counting the bacteria, while the lower intensity values were considered as noise. As bacteria are considered as round or elliptical in microscopy images, the structuring element in all of the following operations was a disk with a fixed size based on the color of the bacterium (disk radius of 8 pixels for the noisier red channel and 7 pixels for the green channel). First the areas with the highest intensities were obtained - a local maximum mask was created based on the dilated image (refer to **Figure 13c**). The mask value was set to True when there was a non-zero value at the overlay of the dilated and thresholded image. This mask showed the location of potential peaks in the image with the background noise.

In order to isolate the peaks, a background mask was created. This mask consisted of boolean values where they were set to True when the pixel values of the thresholded image were 0. To remove an artifact line, the background mask was eroded by the aforementioned structuring element. The mask for the final peaks were determined by using the exclusive OR operation between the local maximum mask and the background mask. In this way some of the noisy True values in the local maximum were set to False. Thereafter, the values in the final peaks' mask were multiplied pixel-wise with the thresholded image, resulting in an image with the brightest intensities detected and each region belonging to a bacterium (refer to **Figure 13d**).

Finally, all of the pixels of values bigger than 0 were set to white and this binarized image was labeled so that every white cluster was counted as a separate bacterium. It ought to be pointed out that this process happened for every plane in a color channel separately. Additionally, in the situation where one bacterium was captured over multiple planes, the bacterium was counted as one. That is, if a white region spanned in the same pixel location throughout multiple planes, the bacteria were counted once. This approach, however, created a bottleneck for the potential cases where multiple bacteria were connected as chains along the depth of the sample.

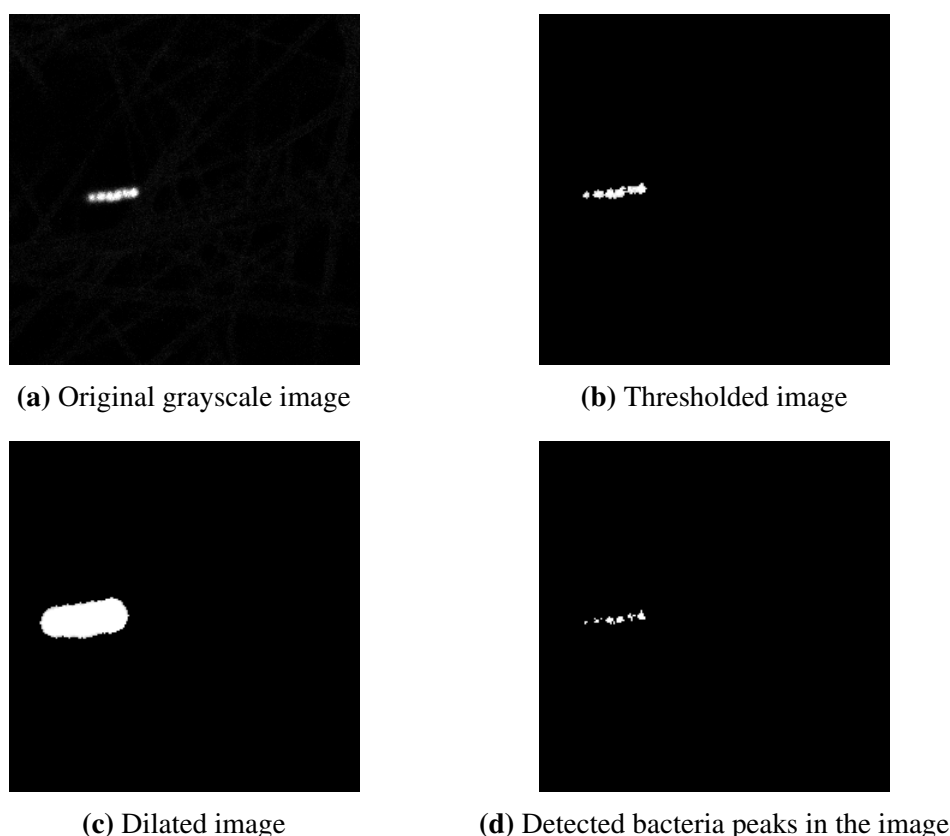


Figure 13. Intermediate results of the bacterial cell analysis pipeline. Here the image plane has been cropped so that only one cluster of bacteria is visible. This process will be repeated throughout all the planes in one stack and the outputs (like in subfigure d) are being overlaid to obtain the amount of bacteria in one sample.

4.3 "FiBar" orchestration, data collection

Both pipelines - the fiber diameter measuring pipeline (with the classical image processing approach) and bacterial cell analysis pipeline - were set up in Windows 10 OS in a lab computer at the Institute of Pharmacy. A task in the Task Scheduler was configured so that it was triggered every 15 minutes indefinitely. The task checks for newly added files in two sub-folders in the lab's common network drive folder. Considering the system's constraints provided in 4.3.1, users can add their microscopy CZI format files and SEM images to the appointed folders ("CZI_input", "SEM_input" under "Automaatika" folder). If a file has been added to any of the sub-folders, the respective pipeline will be triggered. All of the data collected during the procedure will be added to their separate Excel Workbook ("CZI_analysis", "SEM_analysis"). The data collected from the fiber diameter pipeline are the following: file path, diameter measurements, mean, standard deviation,

median and runtime of the pipeline. Information about the file path, the number of stacks in one CZI file, the amount of green and red fluorescent bacteria, and time it took to obtain results from the bacterial cell analysis pipeline are gathered to the "CZI_analysis" workbook. The Excel-based approach was designed to make it easier for the users to use the system compared to having to learn how to prompt values from a database table. In order to alleviate the potential issue of the workbook slowing down, the workbook consists of monthly worksheets. In case the pipeline fails for some reason, it will be restarted as soon as possible. The end-user will know that their image file has been analyzed, once the file from the initial sub-folder is moved to "*_processed" folder where * is either CZI or SEM. Refer to **Figure 14** to grasp an overview of the system orchestration.

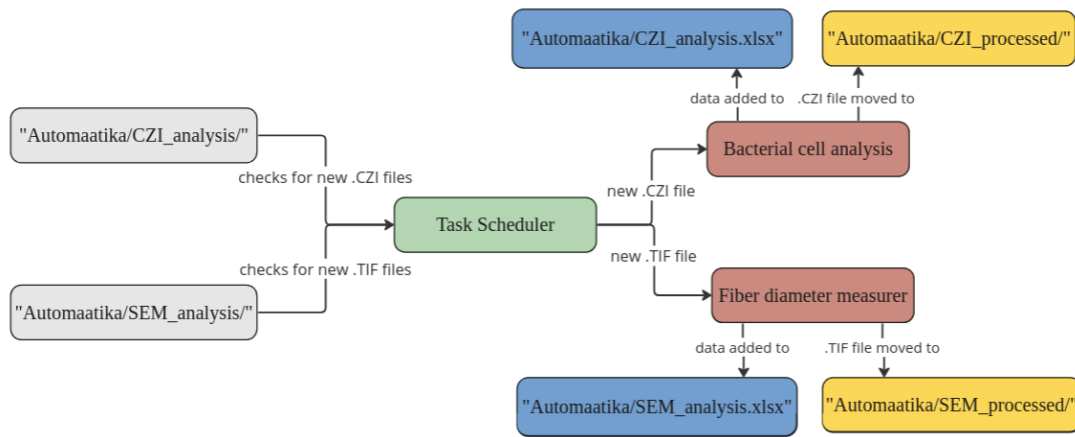


Figure 14. General workflow of "FiBar". The functioning of the tool is governed by the Task Scheduler (green) that checks for new files in the appropriate folders. If a new file (.CZI/.TIF) has been added to a folder (light gray), the respective pipeline (red) is triggered. Once the data have been added to the workbook (nodes in blue), the initial file is moved to the processed folder (nodes in yellow). The edges in this pipeline indicate an action or trigger.

In case the reader wants to run the pipelines, the scripts should be executed in the Python 3.8.10 environment. All of the required libraries and their versions can be installed from the "requirements.txt" file in the thesis Github repository (link provided in the Appendix II. Link to the project repository). Next to having a folder for setting up the tool, a "Quick_runs" folder was created to test the pipelines with single microscopy images (image examples also provided in the repo). Further instructions for testing are provided in the "README.md". Some constraints to consider when running the pipelines are expanded in the next section.

4.3.1 Constraints of "FiBar"

The most important aspects to acknowledge when using the tool are provided below. These were formed based on the input data provided by the Pharmaceutical R&D lab at the Institute of Pharmacy.

For the fiber analysis pipeline, the system assumes the following:

- the SEM input image can be TIF/PNG/JPG format;
- the file name should include "_2k_" or "_5k_" if a magnification of 2k (2000x) or 5k (5000x) was used;
- the OCR part of the pipeline assumes that the value and unit are provided in white, highlighted in black and the horizontal bar is white on a transparent background aligned below the value and unit;
- the scale can have values 1, 2, 3, 10, 200 or 400 and units, nm or um.

In case of the bacterial analysis pipeline, the following is expected:

- the input file is CZI format;
- the user has used red and green detectors to collect the data from the sample;
- staining dyes or fluorescent proteins have been used in the experiment (that have resulted in filled fluorescent bacteria);
- the shape of the raw data has the shape (I,T,C,Y,X) where I (illumination) and T (time) have values of 0;
- the input file can have up to 4 channels, but it needs to includes at least 2 channels for the red and green color.

4.4 Validation of pipelines

The following section describes the methods and data used for validating the pipelines.

4.4.1 Fiber diameter measuring pipeline

The validation of the fiber diameter pipeline was based on research conducted by Stanger et al. and Hotaling et al. [Hotaling et al., 2015, Stanger et al., 2014]. The pipeline was tested on simulated two-tone images (with fibers of constant and varied diameters), as well as actual SEM images of fibers. The simulated images were generated by using mainly Python's libraries NumPy and OpenCV. The synthetic dataset consisted of images (768x1024 resolution) with 30 unordered 2-tone fibers which had diameters of:

- the same size (10-30 pixels);
- three different possible sizes (combinations of 3 diameters with a step of 5 starting from 5 pixels and ending with 35 pixels);
- normally distributed (mean=20 pixels, standard deviation=5 pixels) sizes.

The fibers were either curved or straight lines. As there were 30 fibers in the image, the same amount of measurements were taken from every image automatically. Also, some horizontal bars were generated which imitated the scale component in the SEM images. The generated bars were of varying lengths from 60 to 100 pixels. Altogether there were 30 synthetic images. Refer to **Figure 15** for some examples of the synthetically created images.

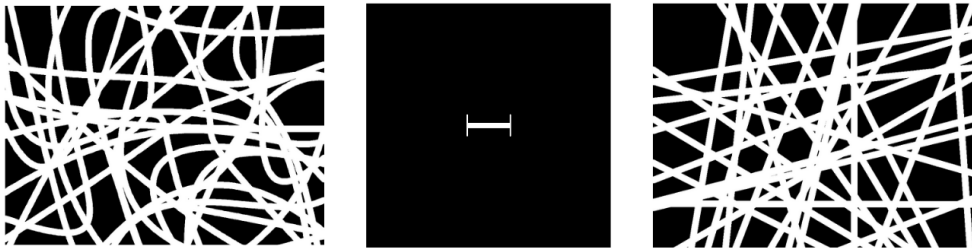


Figure 15. Examples of synthetically created images: the leftmost image shows random curved fibers, whereas, the rightmost image shows generated straight fibers, the middle image is an example of a horizontal bar created.

All of the SEM (Zeiss EVO MA, Germany) images (768x1024 resolution) are courtesy of the Pharmaceutical R&D Lab at the Institute of Pharmacy. The given dataset was divided based on the biological experiments done. The first image set covered 5 biological samples with 33 .TIF images. The images entailed encapsulated *E.coli* or *L.lactis* bacteria in fibers depending on the sample. The SEM images had magnifications of 2k, 5k, 15k and 20k. The dataset of the second experiment consisted of 2 biological samples of chloramphenicol (CAM) loaded fibers. Both samples - control and test - consisted of 3 images with a magnification of 10k. The last dataset consisted of 2 samples with PVA fibers with and without a medication. These samples also had 3 images with 10k magnification.

Based on the examination of the convergence of cumulative average by Stanger et al., it was found that at least 150 measurements per sample would be needed to ensure the reported total average was within the ± 1 pixel error limit [Stanger et al., 2014]. This is also why the data provided by the lab at the Institute of Pharmacy had at least 3 images per sample. Every sample was accompanied with manually measured diameters, leading to a summed total of approximately 300 measurements per sample. Generally, the automatic approach was also set up to gather 300 data points per sample.

This being said, next to manual references (or ground truths), 2 other human experts (who used the ImageJ drawing tool) were included in measuring some of the experiment data. All-in-all images were measured at most by 3 human operators and 2 separate results were evaluated for the automatic approach (U-Net and the classical method approach). In order to assess the goodness of the automatized approach, the resulting measurements were compared to a manual reference distribution-wise as well as statistically. In some cases the distribution-free overlapping index of inter-manual and inter-sample measurements was considered. The distribution-free overlapping index is a normalized measure quantifying similarities (or differences) between empirical distributions, which does not assume the normality of distributions nor any other distributional form. The overlapping aspect is termed from the overlapping area of two distributions [Pastore and Calcagni, 2019].

As for the synthetic 2-tone images, a one-way F-test was used for estimating the significance between the generated and automatically obtained results.

Additionally, the stochasticity of the system and time performance was estimated on an image magnification and automatic approach level.

Almost all of the statistical measures and visualizations were obtained in Python, using the Stats module of Scipy as well as NumPy and Seaborn, Matplotlib. The only exception being the distribution-free overlapping index which was used in R ("overlapping" library).

It was also aimed to compare the custom automatic approach with the ImageJ's plugin DiameterJ (version 1.018). However, upon the time of reviewing, the measuring part did not work properly, so the idea was dropped.

4.4.2 Bacterial cell analysis pipeline

The validation of the bacterial cell analysis pipeline was tested in two ways: through generated images of green and red circles and ellipses and a case study provided by some of the supervisors of this work - Marta Putrinš and Karin Kogermann. While simulated data were used to assess the working of the bacteria counting part of the pipeline, the case study additionally determined the functioning of converting CZI files into image arrays.

The synthetically created images of bacteria (see **Figure 16** for examples) were generated using Python's NumPy and OpenCV. With the amount of points (ranging from 4 to 177) fixed per image, red or green ellipses and circles were generated based on a provided size range (radius range from 3 to 10 pixels). Several images had a couple of colored objects clustered together, sometimes the objects were situated at the edges of the images. Altogether 30 images were generated.

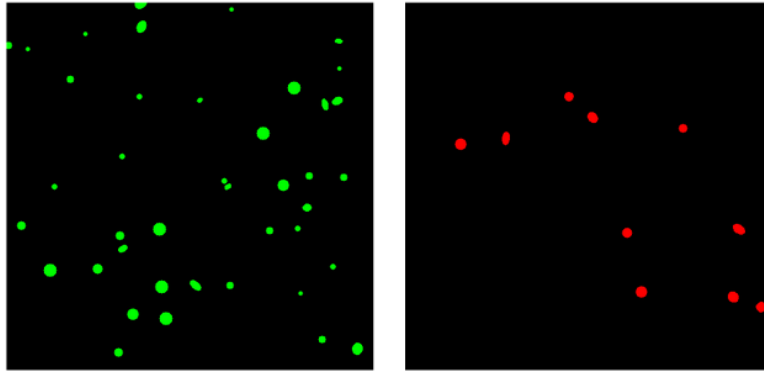


Figure 16. Examples of images with simulated bacteria. The left image contains 41 green circles and ellipses, the right most image shows 11 red circles and ellipses.

The generated images were converted to grayscale, Otsu thresholded and set as input to the counting module. The amount of automatically counted bacteria were then compared to the generated amount. Additionally, the "Analyze Particles" module in ImageJ (version: 1.53t) was tested using the synthetically created images. The thresholding was done using "Auto Threshold" with the default thresholding option and watershedding was done only when there seemed to be clustered objects in the image. The segmented image was set as input for the particle analyzer where default settings were kept for the particle sizes. The counting results were saved and compared to the automatic approach.

The case study provided by the supervisors covered microscopy images of electrospun *L.lactis* bacteria. The microscopy images were captured using the Zeiss LSM 710 confocal microscope. The goal was to answer two research questions using the created bacteria viability pipeline:

1. Does the addition of polyethylene oxide (PEO) have an effect on the fiber permeability?
2. Are bacteria viable within the fiber?

The question whether fiber material had an effect on the results was researched using 24 CZI files of *L.lactis* pre-stained with FM 4-64 (red dye) and after the encapsulation of bacteria, green staining dye (SYTO-9) was added. A copolymer of L-lactide and ϵ -Caprolactone (PLC) as well as combination of PLC and polyethylene oxide (PEO) were polymers used to create the fibers. The hypothesis set by the case study creators was that the addition of PEO to fibers will make the fibers more permeable and more bacteria will stain with SYTO-9. In order to support or reject the null hypothesis, the percentage of SYTO-9 stained bacteria per sample was found for mats with PLC and PLC+PEO.

The answer to the second question was formed based on 25 CZI files where the bacteria were stained with SYTO-9 (green dye) and a nucleic acid stain propidium iodide (PI) (red dye). Both stains were added only after bacteria were encapsulated into ES fiber-mats. The question if bacteria were viable within the fiber was answered through estimating the amount of green (alive) bacteria to the amount of red (dead) bacteria throughout all the planes in one CZI file.

The automated counting results were compared to the manually counted results. Due to the diversity of biological samples, it should be pointed out that a 10% uncertainty range is still considered as successful counting.

All of the statistical measures and visualizations were obtained in Python, using the Seaborn, Matplotlib for visuals and NumPy for calculations.

5 Results and discussion

5.1 Fiber diameter measuring pipeline performance

The following subsections will describe in further detail the different tests that were performed with the fiber diameter measuring pipeline - both on images with simulated as well as with ES fibers.

5.1.1 Horizontal scale bar test

The horizontal scale bar test was a comparison test between two expert measurers and the created algorithm. The experiment tested the algorithm's thresholding approach to detect the scale and compared it with human performance (one line per bar). While the algorithm was able to estimate the correct scale lengths for all the examples, human operators missed the actual length on average by 2 pixels (refer to **Table 2**).

Table 2. Comparison of manual performance and algorithm performance on measuring horizontal bars. The averaged difference compares the average of manual measurements to the true values.

True bar length (px)	60	70	80	90	100
Algorithm's measurement (px)	60	70	80	90	100
Manual measurements (px)	63.01, 61.00	72.33, 73.00	82.01, 76.25	91.34, 93.00	102.01, 102.01
Absolute average difference (px)	2.01	2.67	1.81	2.17	2.01

5.1.2 Testing the diameter measuring algorithm

The diameter measuring algorithm was initially tested on simulated two-tone fibers ("fibers" interchangeably used with the term "lines" here).

The first round of tests was done on images with fibers of a constant diameter within a single image. **Figure 17** shows the measured values for curved and straight lines

(simulating fibers) compared to the generated line diameters. Considering the ratio of line diameter to the image resolution (image resolution stayed constant at 768x1024), images with line diameters in the range of 10-30 pixels are most akin to real SEM images, hence the results for those generated diameters were deemed to be the most important. It can be seen that the error between the generated and measured diameters increases with the size of the generated line diameter. Notably, the median of the measured values is way higher than the generated line diameters of 30 pixels in that image. Additionally, straight lines seem to have a much higher median than curved lines.

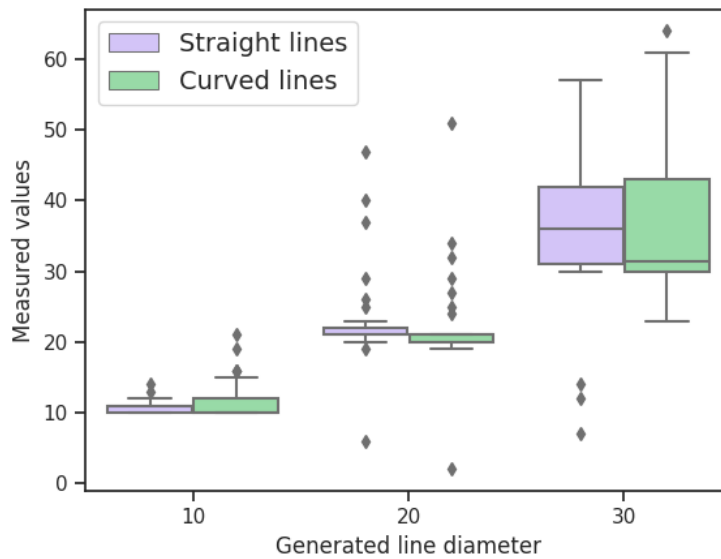
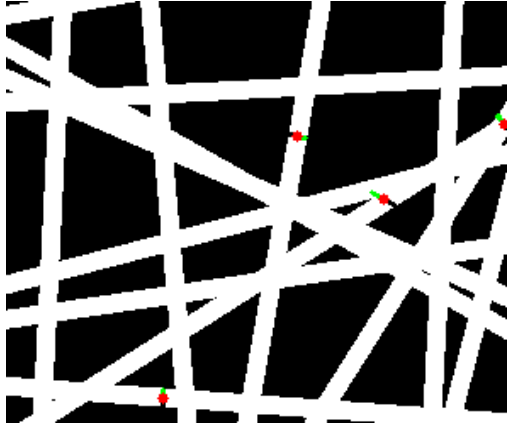
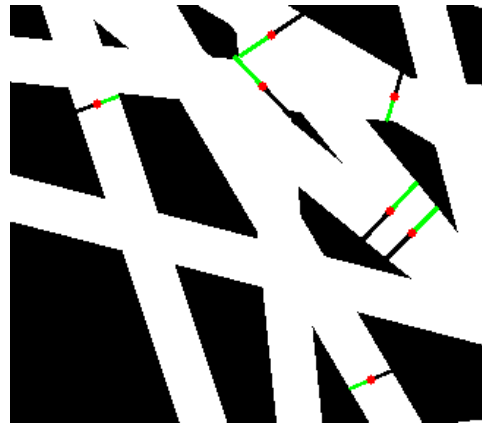


Figure 17. The box plot for automatic measurements for straight and curved lines. The x-axis of the box plot indicates the actual line diameter of the fiber and the y-axis shows the automatically measured values.

This behavior can be explained by visualizing the generated diameters on the original image. Examples of automatic diameter measurement locations on straight-lined fibers are provided by **Figure 18**. With increasing fiber diameter, the likelihood of randomly selecting a point that is part of the fiber crossing increases (as there is less space for other structures). Fiber crossings are V-shaped structures that the algorithm is not really good at avoiding or measuring correctly (see mid-right region in **Figure 18b**). Therefore, the next step to improve this solution is detecting these crossings and removing them before points are randomly selected. One way to determine these crossings is by creating a network from the skeletonized image and removing all intersections of the network's edges. This network-based approach has also been tested in the works of D'Amore et al. [D'Amore et al., 2010].



(a) Straight lines with diameters of 10 pixels.



(b) Straight lines with diameters of 30 pixels.

Figure 18. Measurement locations on the simulated two-tone images. The black segments indicate the path from the starting point to the midpoint (red). The green segments show the path from the midpoint to the end point.

Next, images with 30 simulated fibers (curved or straight lines) of 3 different diameters were experimented with. The goal of having 3 different diameters in one image was to test if multi-modality would become present within the distribution of 100 measurements from the image. While Hotaling et al. generated thousands of measurements per image for the multi-modality test, it was interesting to see if this algorithm could overcome its stochastic effect with fewer iterations [Hotaling et al., 2015]. This did not hold true, as can be seen from the smoothed and normalized distributions of different images in Figures 19a, 19b. It should be noted that whatever the orientation of fibers (curved or straight), larger diameters showed a smaller modality tendency.

A bigger number of measurements was tested as well, in fact, 750 data points were gathered from the same two selected input images. Figures 19c, 19d show the normalized distributions in this case. It is evident that while still not reaching the same probabilities, the peaks are situated in the same diameter range as for the maxima in the estimated distributions of generated values. Hence, the increase in the number of data points has enabled to approach the expected distribution.

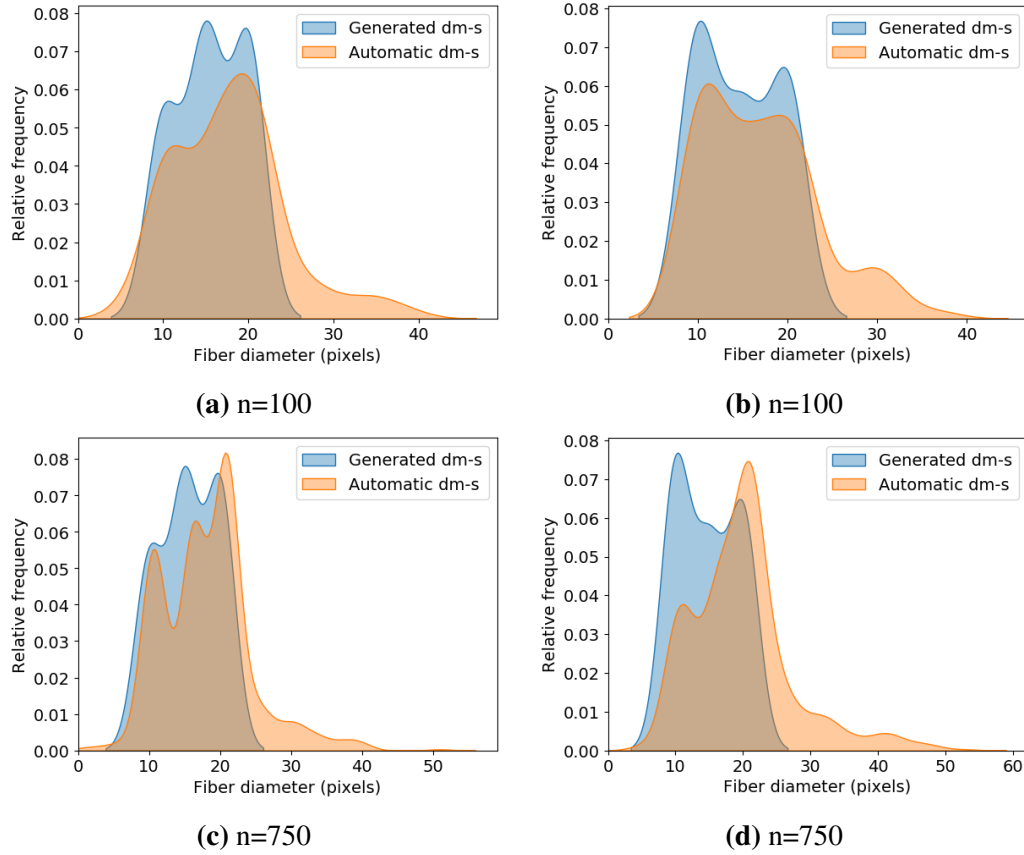


Figure 19. A series of estimated distributions with generated and automatically measured values. The blue distribution represents the generated diameters (30 values), while the orange distribution shows the diameters measured by the algorithm. The input images in subfigures (a) and (c) had three types of straight fibers with diameters of 10, 15, and 20 pixels, while subfigures (b) and (d) had curved fibers with diameters of 10, 15 and 20 pixels. The top row of plots shows orange distributions that were built using 100 data points from the corresponding input image, while the bottom row displays distributions from 750 measurements.

The fiber diameter measurement algorithm was additionally tested on simulated fibers, which diameters were generated from a normal distribution. At first, the diameter measuring algorithm took 30 measurements from every image. 2 sets of manual measurements were also collected from these images. An example of the input image as well as measurement results are shown in **Figure 20**. It can be seen that the medians throughout all 4 sets of measurements are close to each other. The same figure indicates that all of the manual measurements have rather differing value ranges.

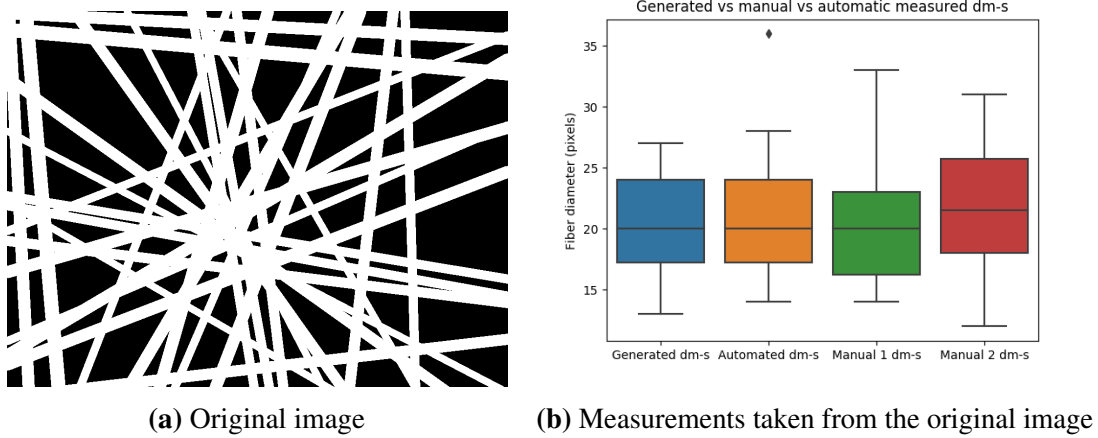


Figure 20. Example of the input image (a) with fiber diameters sampled from a normal distribution. The box plot on the right (b) shows the interquartile range and the minimum/maximum range of each measurement technique. Each box was created using 30 measurement points from the original image.

F-test was performed to validate if the distribution of generated and automatically measured diameters were statistically similar. The null hypothesis was that the means of the automatically measured values are the same as the means of the generated values of that population. Having acknowledged that, 7 different images were tested against this hypothesis and the Bonferroni correction was implemented (at significance level 0.05). Initially the p-value of the one-way F-test was not significant for all except for one sample. However, after increasing the measured data points to 50, the p-value showed that the measured diameters compared to the generated fiber widths were all statistically non-significant. This indicated that there was not enough evidence to reject the null-hypothesis, meaning the two distributions were generated by the same process. It should be pointed out once again that due to the stochasticity of the system, a considerably bigger amount of automatically measured data points is common practice in many works [Hotaling et al., 2015, Götz et al., 2020, Stanger et al., 2014].

Additionally, while looking at images with automatically estimated diameters (examples in **Figure 18**), some aspects became outlined:

- while manually measuring, it is suggested that every fiber is measured only once, this is not the case for the algorithm where some fibers were measured more than once (refer back to **Figure 18b**);
- sometimes the endpoint of the visualized diameter did not terminate in the background, but somewhere in the fiber. This is not a distance measuring error but the issue stems in the way the endpoint was chosen for visualization. The endpoint

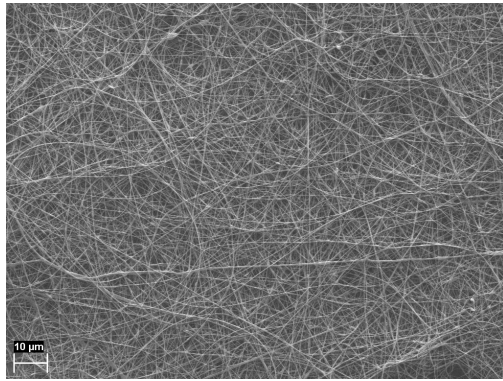
follows the initial direction provided at the starting point while the distance from the midpoint is accurate.

5.1.3 SEM image measurement tests

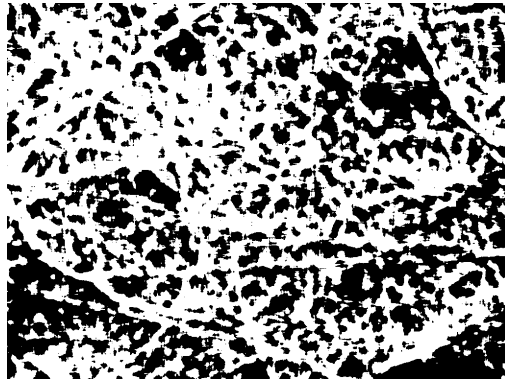
The complete pipeline (segmentation and the measuring algorithm) was tested on images from three different experiments. The pipeline had two sets of results obtained using segmentation approaches: U-Net and classical, that were compared to the provided manual (ground truth) measurements.

During the first experiment, it became evident that for magnifications of 2k and 5k, the diameter results were very extreme, sometimes leading to values over 4000 nm. Hence, some adjustments were made. The most alarming aspect was how the image was segmented for 2k and 5k images (see **Figure 21** for comparison of original **Figure 21a** and its segmentations **Figure 21b**, **Figure 21c**). Both segmentations had lost a lot of the finer details of fibers from the original image. Considering that the algorithm detects the distance to the midpoint within the white regions, it is of no surprise that in the case of vast white areas, the diameter values were so much bigger than expected. Two different attempts were made to improve this:

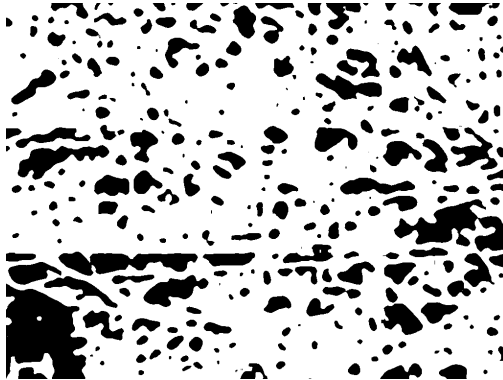
- decrease the size of subimages that were segmented by U-Net: as the model was trained on a 15k magnification image, the model is used to predict on a 256x256 subimage with a specific amount of detail, however a smaller magnification (2k, 5k) means there are a lot more features in one subimage. Hence, taken into account the $n=15/\text{smaller-magnification-value}$ ratio, the image patches were decreased by rounded n times (e.g., for 2k the subimage size will be $256/8 = 32 \times 32$). The prediction did become more detailed, yet still not very promising, an example of an improved segmented image is provided in **Figure 21d**.
- simplifying the segmentation approach to just thresholding: it seemed that the foreground fibers were more uniform in color, unlike higher magnification images with larger fibers where the intensities within one fiber change quite a lot. This meant it was possible to test out the global Otsu's method, which was relatively successful in terms of maintaining more of the original detail. This was also supported by the more sensible x-axis value range of the new histogram (**Figure 21e**, **Figure 21f**). The histogram has small number of bins, likely because of the amount of detail the segmentation omits.



(a) Original 2k SEM image



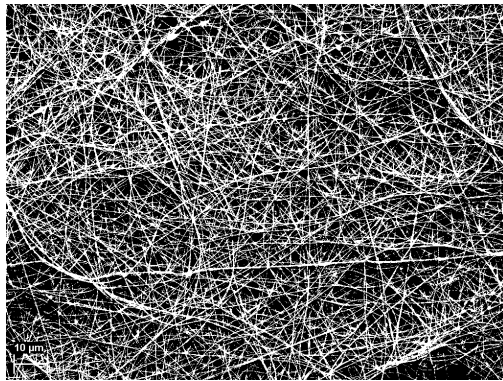
(b) Classically segmented image



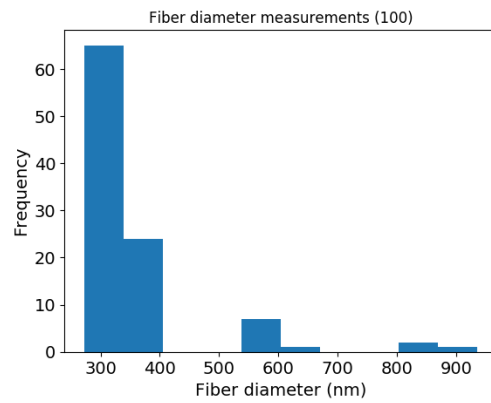
(c) U-Net segmented image



(d) Improved U-Net segmented image



(e) Otsu segmented image



(f) Histogram of the Otsu segmented image

Figure 21. Different segmentation methods tested on a SEM image with 2k magnification.

Often, experts themselves do not measure fibers from 2k and 5k magnifications due

to the likelihood of a cumulative measuring error. Images with these magnifications are often created to visually assess the characteristics of the imaged sample. This is why, the comparison between the automatic and manual approach is considering images with magnification other than 2k and 5k. Nonetheless, if the 2k or 5k magnified images are to be used more, the author suggests to train multiple U-Net models for images of different magnifications.

Table 3 shows the statistical comparison of diameter measurements from the first experiment's images with 15k and 20k magnifications. Except for sample no 4, it is evident that the automatic approaches tend to provide bigger values on average than manual. Initially, it was hypothesized that the issue lies in the final distance calculation, where the covered distance to the midpoint is multiplied by 2. This could be a matter of a few pixels, but as it was seen in **Figure 18**, the algorithm was finding the diameters correctly. Yet, even a deviation of a few pixels from the ideal diameter (perpendicular to the fiber edge), may result in many nanometers depending on the provided scale bar of course. For instance, a scale bar of 70 pixels is considered as 2 μm (2000 nm) in reality, this means that every pixel is around 29 nm-s. Thus, even minor differences in the measuring process can lead to big changes in the summarized outcome. This is also potentially the reason why the standard deviation is much bigger for the automatic methods. However, this could also be a sign of bias on the behalf of the human expert, as it indicates a more narrow coverage of the diameter values.

Additionally, it is worthy to mention that the median should be considered more as a global statistical measure in the fiber diameter measuring community, as the average and standard deviation assume the normal distribution, which is often not the case with the amount of measurements taken.

Table 3. Comparative statistics of the manual and automatic performance from the data of the first experiment. Each sample had approximately 300 measurements. Samples no 1, 3 and 4 contained images of 20k magnification, samples no 2 and 5 had images of 15k magnification.

Method	Sample no	Mean (nm)	Standard deviation (nm)	Median (nm)
Classical segmentation	1	305	170.3	270
Manual reference		242	76.2	225
U-Net		338	175.9	292
Classical segmentation	2	293	156.7	268
Manual reference		240	138.2	205
U-Net		272	103.5	238
Classical segmentation	3	287	137.6	250
Manual reference		219	72.3	207
U-Net		288	122.1	238
Classical segmentation	4	602	251.1	563
Manual reference		539	78.7	539
U-Net		442	218.5	457
Classical segmentation	5	308	136.4	273
Manual reference		274	99.8	248
U-Net		301	114.6	273

Understanding how much of a difference between the measurements of the automatic and the manual approach is too vast was estimated using the distribution-free overlaying index (DFOI). **Table 4** is an example of how the similarity indices were compared pairwise between the methods. The higher the index value, the more similar the data distribution is between the methods. It indicates that the distributions between the Classical and U-Net methods are rather similar but dissimilar compared to the manual dataset.

Table 4. Distribution-free overlaying index table of methods' measurement distributions from Sample no 1 images of the first experiment. Each distribution had approximately 300 data points.

	U-Net	Classical	Manual
U-Net	1	0.71	0.56
Classical	0.71	1	0.49
Manual	0.56	0.49	1

As there is no ground truth in this field and the uncertainty of the automatic approaches remain, the similarity between the human experts was investigated. This called out for the second experiment where fibers were measured by 2 people next to the manually measured data provided by the lab (will be named Reference in the table below). The DFOIs between both automatic methods and manual measurements of one sample are provided in **Table 5**.

Table 5. Distribution-free overlaying index table of methods' measurement distributions from Sample no 1 images of the second experiment. Each distribution had approximately 300 data points.

	U-Net	Classical	Reference	Manual 1	Manual 2
U-Net	1	0.83	0.57	0.42	0.72
Classical	0.83	1	0.66	0.81	0.44
Reference	0.57	0.66	1	0.43	0.67
Manual 1	0.42	0.81	0.43	1	0.51
Manual 2	0.72	0.44	0.67	0.51	1

It can be seen above that the similarity index between the manual distributions are not really homogeneous: Manual 1 and Manual 2 have similarity of 0.43 and 0.67 to

the Reference respectively. Automatic approaches reached analogous similarity indices when compared to the Reference distribution. In general, the data produced by different approaches are not really similar except for the Classical and U-Net approach. The same pattern was observed in the other sample of the second experiment.

At this point, it is worthy to remind the reader about the general goal of measuring fiber diameters - to assess how different alterations (addition of drug/bacteria to the polymer, environmental changes) in the ES process reflect in the fiber mat diameter distribution. Thus, the difference between distributions for any method should be at minimum the same signed deviation. Another potential aspect to consider is the inter-sample similarity - where the similarities between the samples of one method should be in the same range compared to the other methods. Both of these outlooks were tested in the third experiment. Below is the conclusive table for both statistical measures as well as the inter-sample DFOIs for the data from the third experiment **Table 6**.

Table 6. DFOI and statistics table of samples' measurement distributions from all the methods of the third experiment. Each distribution had approximately 300 data points. The Manual Reference is the data provided by the lab, Manual 1 method signifies the additional measurements done by a human.

Method	Sample no	DFOI	Mean (nm)	Standard deviation (nm)	Median (nm)
Classical segmentation	1	0.69	857	443.9	752
Manual reference		0.58	596	166.8	573
U-Net		0.8	727	312.7	695
Manual 1		0.55	716	283	677
Classical segmentation	2	1	718	303.6	638
Manual reference		1	541	93.1	536
U-Net		1	777	293	721
Manual 1		1	614	117.3	606

In the third experiment, Sample no 2 was different from Sample no 1 by having an additive in the polymer which theoretically should decrease the fiber diameter. This theoretical note was proven by all the medians of the measurement data of different methods, as all of the medians did decrease in the second sample (same signed change in **Table 6**). It is evident that the inter-sample DFOI between manual methods is very

similar, reaching values of 0.55 and 0.58. The classical method was closer to the manual methods DFOI-wise than the U-Net approach. Based on this result, the classical image processing approach was opted to be the segmentation method in the "Fibar" tool. The standard deviations and means were way bigger than the manual reference, so it is reasonable to take a look at what was happening in the background. **Figure 22** shows an original image from the third experiment, the segmented versions and the original image with diameter lines.

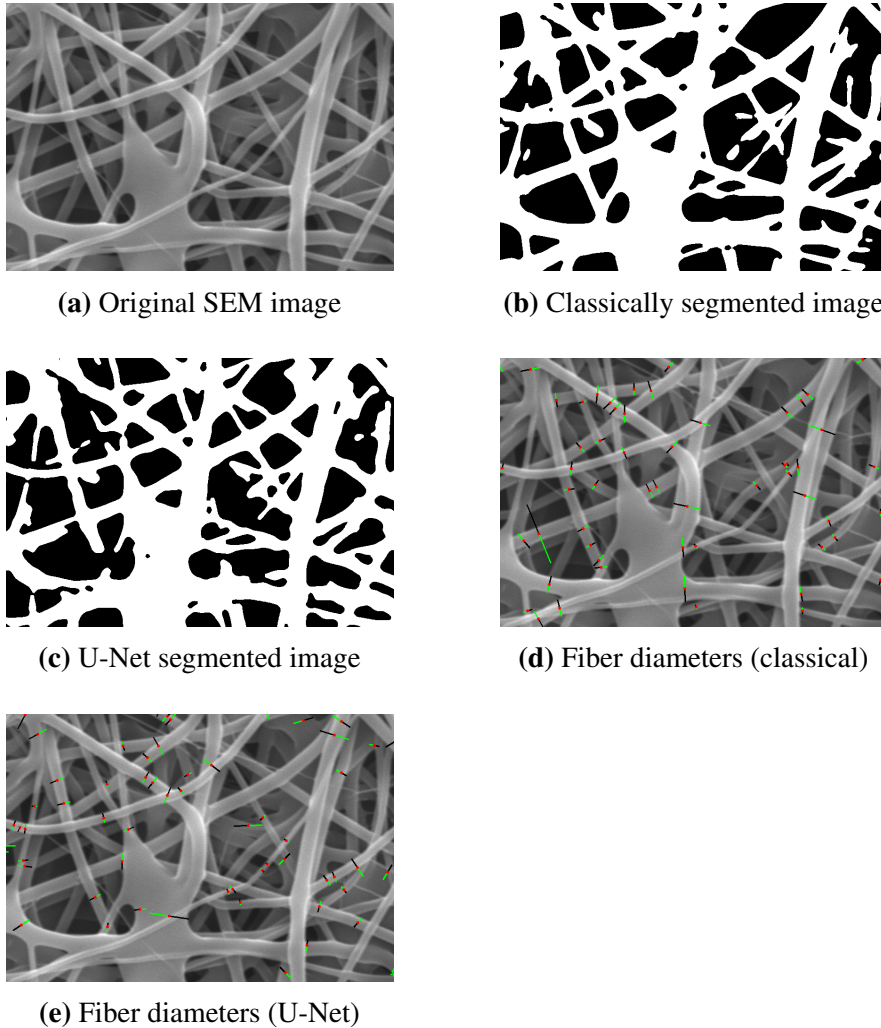


Figure 22. The different stages of the fiber diameter pipeline - original image (a), the segmented versions (b, c) and the fiber diameter measurements (d, e) on the original image. All of the images have been cropped by 100 pixels from all edges. In subfigures d, e, the black lines indicate the path from the starting point to the midpoint (red). The green lines show the path from the midpoint to the end point.

From the figure above, it can be seen that in some regions (e.g. bottom-right corner) the U-Net approach (**Figure 22c**) is better at classifying foreground fibers than the classical approach. At the same time, some fibers are not fully connected in the U-Net prediction. Hence, both approaches seem to be visually on par. However, the quality of segmentation is what determines the goodness of the fiber diameter measurements. It can be seen that the segmentation step should be further enhanced to improve the measurement accuracy, as some diameter are definitely not measuring the foreground fiber (**Figure 22d**). It can also be observed that sometimes the diameter of the line is not being measured perpendicular to the fiber. This is reasoned through the approach the line to the mid-point is created. As a reminder, for finding distance, the closest white point on the medial axis (mid-point) from the start-point is selected. However, once the diameter line is determined, the perpendicularity with the edge of the fiber is not considered. So, there are cases when the diameter line orthogonalizes with the fiber edge and there are cases when the lines are not so perpendicular. This is why some diameters are not crossing the fibers at a right angle.

In image analysis applications, performance time has often been considered as a quality metric. Although time was not as critical for the simulated fibers (30 measurements), it is not the case when doing 300 measurements per sample. It took about 5-7 minutes to measure 100 diameters manually from one image, whereas for the system this value ranged between 0.75 and 1.75 minutes (Intel i5 8th gen processor) depending on the level of detail in the image. There was an evident time difference between the U-Net and classical image processing approach. The averaged cumulative time cost for one set of images of both methods is provided below, in **Figure 23**.

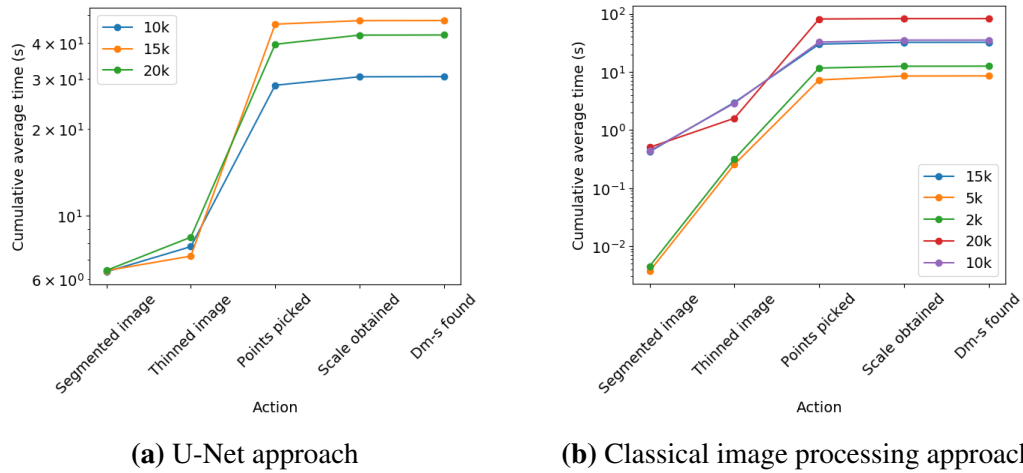


Figure 23. Averaged (10 images per magnification) cumulative time in the diameter measuring pipeline. The x-axis shows specific actions of the fiber diameter measuring pipeline as timestamps. The y-axis is scaled logarithmically for time. The time for 2k and 5k magnifications were only measured using the classical image processing method.

This shows that on average it took about 6 seconds to segment a 768x1024 image using U-Net, while it took less than a second to segment the images of the same magnification classically. It can be also seen that the starting point selection was the most time consuming for both methods, taking a considerably more amount of time for the classical image approach. It is likely to be due to the added complexity the classical approach adds to the segmented image. Despite the method, this part of the pipeline could be improved by adopting a sub-region-based approach. So far, the pixel locations are randomly selected throughout the whole image and for every location the suitable conditions are checked. In order to guarantee that the pixels are selected from the whole image uniformly, the image can first be split into four sub-regions. From each sub region a specific amount of random locations tested against conditions - this reduces the search space for the random iterator.

Next, it was of interest to determine how the random behavior of point selection affects the output results in the fiber diameter measuring pipeline. This being said, the randomness of the point selection was put to the test by iterating the fiber diameter measuring process (classical approach) 10 times for the same images and comparing distributions of each iteration. Altogether 4 images with unique magnifications were selected from different experiments. **Figure 24** displays the iterations' distributions of images with different magnifications.

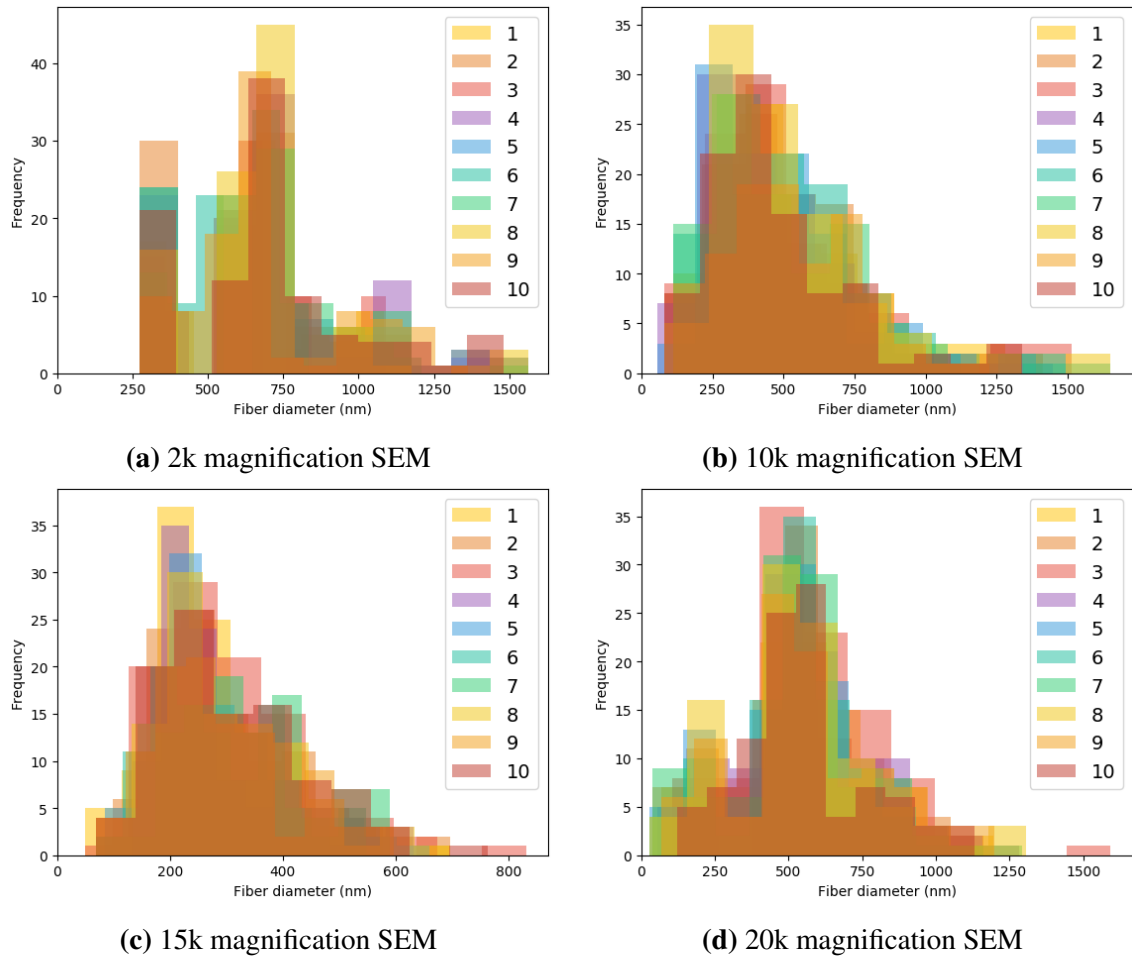


Figure 24. Diameter measurement histograms from images of different magnifications. Diameters from one image were measured 10 times with different starting points at each iteration. The dark orange in the subplots indicates the overlay of all 10 iterations' measurements.

While some minor changes are visible, no outlying peaks or aberrations occur within the 10 iterations of fiber diameter measuring. Thus, it can be assumed that the fiber diameter pipeline is robust.

5.2 Validation of the bacterial cell analysis pipeline

The validation of the bacteria analysis pipeline is divided into two tests based on the input data - synthetic data as well as real-life microscopy images in CZI format. The first test allows to evaluate the goodness of the bacteria counting module of the pipeline, the case study proves the functioning of the whole pipeline.

5.2.1 Simulated green and red bacteria

The images of simulated green and red bacteria were converted to grayscale, thresholded and set as input to the bacteria counting part of the pipeline. The measurements were then compared with the outputs obtained from the Particle Analyzer in ImageJ. The results are provided in **Figure 25**. The average percent-error for the ImageJ tool was 4.5%, while the created counting approach had an average error of 5.3%. This being said, the process of analyzing the images in ImageJ one-by-one was time-consuming and rather mundane.

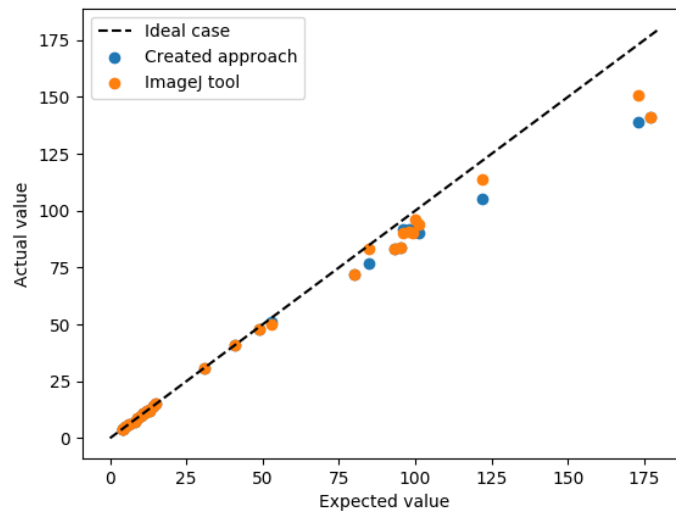


Figure 25. Automated counting of simulated red and green bacteria (values in blue) compared with the results from the ImageJ Particle Analyzer tool (values in orange). Most of the ImageJ values overlay with the created approach, hence some dots are not visible. The black dashed line resembles the ideal case where all of the particles are counted correctly.

It can be seen from **Figure 25** that while the counter is pretty accurate for small amounts of circles/ellipses in the image, while increasing the amount, the counter seems to miss a few circles/ellipses. There can be multiple reasons for this, one of them is that

the size of the morphological operation kernel is fixed based on the bacteria sizes in the case-study. This creates the need to adaptively modify the size of the disk based on the input image. This could be solved by prompting the user for a numerical parameter, such as the average area per bacterium in pixels. This user-based calibration feature is supported by multiple applications, like the previously mentioned CellProfiler [McQuin et al., 2018]. Alternatively, the disk size could be determined by detecting contours on the image and a median contour area is selected to describe the bacteria size in the input image. This approach understandably assumes a segmented image with as little noise as possible, as too much noise may affect the median value.

5.2.2 Case study for bacterial cell analysis

The case study consisted of two research question, to remind the reader, an overview of the study for bacterial cell analysis is provided in **Table 7**.

Table 7. Case study structure and setup.

Dataset	Research topic	Bacteria	Additional staining after ES
24 CZI files (14 with PEO)	Effect of PEO on the fiber permeability	<i>L.lactis</i> pre-stained with FM 4-64 (red dye)	SYTO-9 (green dye)
25 CZI files (7 with 24h)	Bacteria viability in fiber	<i>L.lactis</i>	SYTO-9 (green dye) + PI (red dye) added at 0th hour or 24th hour

The first research question focused on determining if the addition of PEO to the fiber polymer would improve permeability, in other words, determining if the amount of SYTO-9 stained (green fluorescent) had increased in the presence of PEO. **Figure 27** shows that in general the addition of PEO ("Fibers with PLC+PEO") to PLC does result in more green stained bacteria compared to fibers with just PLC. The median proportion of green bacteria in the samples with PEO was approximately 40%, whereas for the samples without PEO, the result was around 60%. The zero values in the plot are caused either because no signal was found in any of the channels, or there were no green dye stained bacteria in the sample.

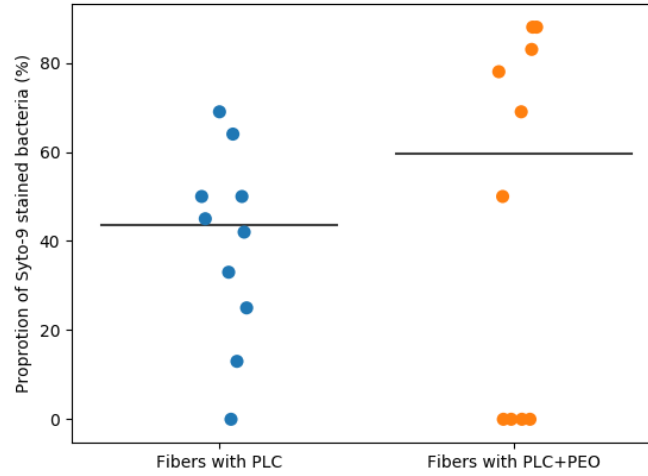


Figure 26. Proportion of viable (SYTO-9 stained) bacteria in the samples with PLC compared with samples with PEO+PLC.

The effect of PEO as conditional distributions between green and red bacteria are visualized in Figure 27.

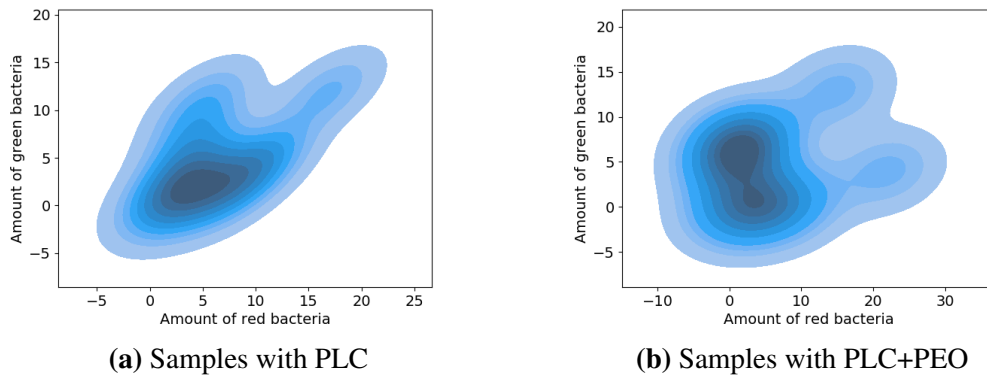


Figure 27. Conditional distributions of red and green bacteria in samples with PLC or combination of PLC+PEO. The x-axis shows the estimated amount of red bacteria, while the y-axis displays the estimated amount of green bacteria. The negative values are apparent because the estimations were not clipped from 0 but can be omitted in this regard.

It can be seen that the conditional distribution with PLC samples (27a) is situated more along the horizontal axis, while the sample with PLC+PEO (27b) is more likely to stretch along the vertical axis. This means that the likelihood of having a bigger amount of green bacteria is higher in PLC+PEO samples compared to samples without PEO. This

supports the initially set null hypothesis that the addition of PEO improves permeability of ES fibers.

The second dataset consisted of samples with SYTO-9 and PI stained bacteria, where PI (red dye) was a signal of dead bacteria. PI and SYTO-9 were added to the fiber mat either at the 0th hour after ES or 24 hours after ES. In the latter case, the mat was incubated in bacteria growth enabling conditions. The second research question tried to determine if the bacteria were viable within the fiber, i.e. evaluate the amount of green bacteria (SYTO-9 stained) to dead bacteria (PI-stained). **Figure 28** shows results for viable bacteria in samples.

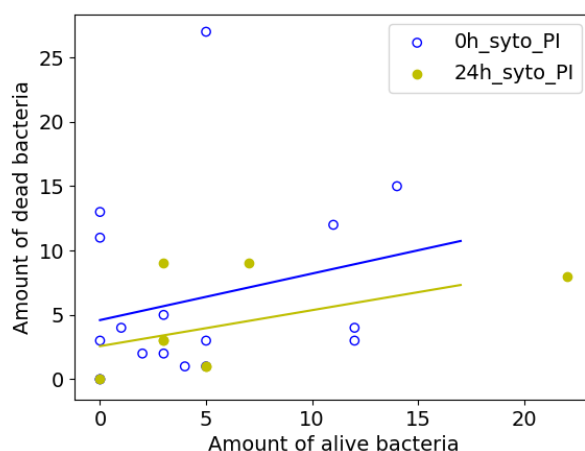


Figure 28. Amount of viable and dead bacteria in the SYTO-PI samples. Each set ("0h_syto_PI", "24h_syto_PI") consisted of multiple sample images. The set "0h_syto_PI" contained sample images taken right after the staining. The set "24h_syto_PI" involved sample images of bacteria which were stained after 24h of incubation in growth enabling conditions. The blue and yellow regression lines were fitted on the 0h and 24h samples' data.

As can be seen from **Figure 28**, the "24h_syto_PI" samples tend to follow a trend for having more alive bacteria in the sample than dead bacteria. On the contrary, the "0h_syto_PI" samples seem to have a smaller proportion of alive bacteria. On average 36% of green bacteria were found in samples of the set "0h_syto_PI", however, after 24h of fiber mat incubation and then staining the average proportion had increased to 51%. This indicates that bacteria are able to remain viable in the fiber for a longer period of time. On average there were 5 green bacteria per sample in the set of "0h_syto_PI", while approximately 6 green bacteria per sample in the "24h_syto_PI" set.

When comparing automatic results with visual observations, the proportions of green bacteria in all samples were correctly determined, however the discrete amounts of

bacteria varied slightly. This could be improved by restructuring the bacterial analysis pipeline in such a way that the mean intensity value belonging to a bacterium in the image is compared between the red and green channel. The channel with the higher intensity would be set as the color of the bacterium. This approach would require a better segmentation method, as the current version cuts off quite a lot of the signal in the channel with low intensity values. This leads to having no signal in one channel, which is not true in the case of actual images where even visually red bacteria have small intensity values in the green channel. Additionally, the current solution contains multiple parameters, like the threshold value, disk size for the morphological operations, that could be more adaptive based on the input data. On top of the contour detection suggestion provided in the synthetic test, the metadata of the CZI could be put into more use. For example, the slice size of the physical sample could be useful to determine the bacteria count per volume which would provide better insight of the bacteria location in 3D space.

It should also be pointed out that the trends for both research questions were established, yet the statistical evidence to support these tendencies is missing. Hence more microscopic data should be generated which if added to the suitable system's input folder ("CZI_input") can now be analyzed in a matter of seconds.

6 Conclusions

The main result of this work was the creation of "FiBar": a tool for analyzing microscopy images of complex biomaterials. Unlike no other solution known to the author, the tool is a suitable fit for evaluating the quality of fibrous material as well as the analysis of encapsulated bacteria in the material. The tool consists of two pipelines, one for measuring fiber diameters in SEM images and the other for analyzing bacterial cells in Zeiss microscopy images.

Next to providing methods that coined "FiBar", this work also presented ways on how to improve the tool. In terms of the fiber measuring pipeline, the segmentation should be advanced further as it is the input for the measuring module. One way to consider, is to train separate U-Net models for specific magnifications. The point picking process could possibly be accelerated by selecting random points from sub-regions, thereby guaranteeing a uniform coverage of the sample. The measuring process could be improved by considering the angle between the fiber edge, where the measurement is taken, and the diameter line. Additionally, the erroneous V-shaped structures can be omitted by detecting and removing them before the starting points are selected. In case of the bacterial cell analysis pipeline, the fixed parameters in image processing could be made more adaptive, either by utilizing metadata from the CZI file or applying some image processing method (such as contours) to obtain information about the characteristics of the image.

"FiBar" has been successfully setup in the lab computer (Windows OS) at the Institute of Pharmacy. It requires no other input from the user besides a file of suitable format (CZI or TIF/PNG/JPG) in the designated folder. It was proved in the Results section that both pipelines can be considered as viable alternatives for the manual analysis. Thus, "FiBar" awaits to create value out of new microscopy images by providing a better understanding about the samples and their possible behavior in real applications.

7 Acknowledgements

There are multiple people the author would like to acknowledge who were of great support during this invigorating experience. The biggest gratitude goes towards all of the supervisors - the emotional support, weekly meetings and input data are what kept the author afloat. A big thanks goes to the IT Support Service staff at the University of Tartu, especially Enno Kaasik who helped make sense of the temper tantrum Task Scheduler was having. The author would also like to express her appreciation towards the colleagues who were inspirational on a day-to-day basis. If the reader feels left out then not to worry, it is still somewhat likely you have encouraged the author in one way or another.

References

- [Pai, 2016] (2016). Oufi: An integrated software package for high-accuracy, high-throughput quantitative microscopy analysis. *Molecular Microbiology*, 99:767–777.
- [aic, 2020] (2020). *AICSImageIO*. Allen Institute for Cell Science. Accessed April 15, 2023.
- [pyt, 2023] (2023). Pytesseract 0.3.10. <https://pypi.org/project/pytesseract/>. Accessed March 6, 2023.
- [Bartyzel, 2015] Bartyzel, K. (2015). Adaptive kuwahara filter. *Signal, Image and Video Processing*, 10(4):663–670.
- [Begg and Berlin, 1988] Begg, C. B. and Berlin, J. A. (1988). Publication bias: A problem in interpreting medical data. *Journal of the Royal Statistical Society. Series A (Statistics in Society)*, 151(3):419.
- [Berg et al., 2019] Berg, S., Kutra, D., Kroeger, T., Straehle, C. N., Kausler, B. X., Haubold, C., Schiegg, M., Ales, J., Beier, T., Rudy, M., Eren, K., Cervantes, J. I., Xu, B., Beuttenmueller, F., Wolny, A., Zhang, C., Koethe, U., Hamprecht, F. A., and Kreshuk, A. (2019). ilastik: interactive machine learning for (bio)image analysis. *Nature Methods*, 16:1226–1232.
- [Bray et al., 2015] Bray, M. A., Vokes, M. S., and Carpenter, A. E. (2015). Using cellprofiler for automatic identification and measurement of biological objects in images. *Current Protocols in Molecular Biology*, 2015:14.17.1–14.17.13.
- [Chen et al., 2010] Chen, X., Jiang, Y., Shen, W., and Han, G. (2010). Research on fiber diameter automatic measurement based on image detection. volume 7656, page 76563L. SPIE.
- [Chiang et al., 2015] Chiang, P. J., Tseng, M. J., He, Z. S., and Li, C. H. (2015). Automated counting of bacterial colonies by image analysis. *Journal of Microbiological Methods*, 108:74–82.
- [Collins, 2007] Collins, T. J. (2007). ImageJ for microscopy. *BioTechniques*, 43(1S):S25–S30.
- [D’Amore et al., 2010] D’Amore, A., Stella, J. A., Wagner, W. R., and Sacks, M. S. (2010). Characterization of the complete fiber network topology of planar fibrous tissues and scaffolds. *Biomaterials*, 31:5345–5354.

- [Davis et al., 2005] Davis, M. E., Hsieh, P. C., Grodzinsky, A. J., and Lee, R. T. (2005). Custom design of the cardiac microenvironment with biomaterials. *Circulation Research*, 97(1):8–15.
- [Ducret et al., 2016] Ducret, A., Quardokus, E. M., and Brun, Y. V. (2016). Microbej, a tool for high throughput bacterial cell detection and quantitative analysis. *Nature Microbiology*, 1.
- [Eliceiri et al., 2012] Eliceiri, K. W., Berthold, M. R., Goldberg, I. G., Ibáñez, L., Manjunath, B. S., Martone, M. E., Murphy, R. F., Peng, H., Plant, A. L., Roysam, B., Stuurmann, N., Swedlow, J. R., Tomancak, P., and Carpenter, A. E. (2012). Biological imaging software tools.
- [Environment, 2018] Environment, O. M. (2018). *OME Data Model and File Formats 5.6.3 documentation*. Open Microscopy Environment. Accessed March 26, 2023.
- [Fernando Cossio, 2023] Fernando Cossio, Yael Suarez, L. C. A. d. L. (2023). Fiber-diameter. <https://www.FiberDiameter.com>. Accessed March 26, 2023.
- [Giusti et al., 2018] Giusti, R., Zanini, F., and Lucchetta, G. (2018). Automatic glass fiber length measurement for discontinuous fiber-reinforced composites. *Composites Part A: Applied Science and Manufacturing*, 112:263–270.
- [GmbH, 2011] GmbH, C. Z. M. (2011). *ZISRAW (CZI) File Format*. Carl Zeiss MicroImaging GmbH. Accessed March 26, 2023.
- [Gohlke, 2023] Gohlke, C. (2023). *Read Carl Zeiss(r) Image (CZI) files*. Accessed April 10, 2023.
- [Götz et al., 2020] Götz, A., Senz, V., Illner, S., and Grabow, N. (2020). Computed fiber evaluation of SEM images using DiameterJ. *Current Directions in Biomedical Engineering*, 6(3):438–441.
- [Haik et al., 2017] Haik, J., Kornhaber, R., Blal, B., and Harats, M. (2017). The feasibility of a handheld electrospinning device for the application of nanofibrous wound dressings. *Advances in Wound Care*, 6(5):166–174.
- [Hartmann et al., 2021] Hartmann, R., Jeckel, H., Jelli, E., Singh, P. K., Vaidya, S., Bayer, M., Rode, D. K. H., Vidakovic, L., Díaz-Pascual, F., Fong, J. C. N., Dragoš, A., Lamprecht, O., Thöming, J. G., Netter, N., Häussler, S., Nadell, C. D., Sourjik, V., Kovács, Á. T., Yildiz, F. H., and Drescher, K. (2021). Quantitative image analysis of microbial communities with BiofilmQ. *Nature Microbiology*, 6(2):151–156.

- [Hartmann et al., 2020] Hartmann, R., van Teeseling, M. C., Thanbichler, M., and Drescher, K. (2020). Bacstalk: A comprehensive and interactive image analysis software tool for bacterial cell biology. *Molecular Microbiology*, 114:140–150.
- [Hotaling et al., 2015] Hotaling, N. A., Bharti, K., Kriel, H., and Simon, C. G. (2015). Diameterj: A validated open source nanofiber diameter measurement tool. *Biomaterials*, 61:327–338.
- [Huang and Wang, 1995] Huang, L.-K. and Wang, M.-J. J. (1995). Image thresholding by minimizing the measures of fuzziness. *Pattern Recognition*, 28(1):41–51.
- [ImageJ, 2023] ImageJ (2023). Particle analysis. <https://imagej.net/imaging/particle-analysis>. Accessed May 2, 2023.
- [Kamble et al., 2017] Kamble, P., Sadarani, B., Majumdar, A., and Bhullar, S. (2017). Nanofiber based drug delivery systems for skin: A promising therapeutic approach. *Journal of Drug Delivery Science and Technology*, 41:124–133.
- [Klein et al., 2009] Klein, S., Kuhn, J., Avrahami, R., Tarre, S., Beliaevski, M., Green, M., and Zussman, E. (2009). Encapsulation of bacterial cells in electrospun microtubes. *Biomacromolecules*, 10(7):1751–1756.
- [Lanno et al., 2020] Lanno, G.-M., Ramos, C., Preem, L., Putrinš, M., Laidmäe, I., Tenson, T., and Kogermann, K. (2020). Antibacterial porous electrospun fibers as skin scaffolds for wound healing applications. *ACS Omega*, 5(46):30011–30022.
- [Liu and Hwang, 2012] Liu, H. L. and Hwang, W. R. (2012). Permeability prediction of fibrous porous media with complex 3d architectures. *Composites Part A: Applied Science and Manufacturing*, 43(11):2030–2038.
- [Matrix, 2023] Matrix, P. W. (2023). The technology. <https://www.renovoderm.tech/Research/Our-Technology/>. Accessed May 6, 2023.
- [McQuin et al., 2018] McQuin, C., Goodman, A., Chernyshev, V., Kametsky, L., Cimini, B. A., Karhohs, K. W., Doan, M., Ding, L., Rafelski, S. M., Thirstrup, D., Wiegand, W., Singh, S., Becker, T., Caicedo, J. C., and Carpenter, A. E. (2018). Cellprofiler 3.0: Next-generation image processing for biology. *PLoS Biology*, 16.
- [Meinel et al., 2001] Meinel, L., Illi, O. E., Zapf, J., Malfanti, M., Merkle, H. P., and Gander, B. (2001). Stabilizing insulin-like growth factor-i in poly(d, l-lactide-co-glycolide) microspheres. *Journal of Controlled Release*, 70(1-2):193–202.
- [Montero and Lang, 2012] Montero, A. S. and Lang, J. (2012). Skeleton pruning by contour approximation and the integer medial axis transform. *Computers & Graphics*, 36(5):477–487.

- [Moore et al., 2015] Moore, J., Linkert, M., Blackburn, C., Carroll, M., Ferguson, R. K., Flynn, H., Gillen, K., Leigh, R., Li, S., Lindner, D., Moore, W. J., Patterson, A. J., Pindelski, B., Ramalingam, B., Rozbicki, E., Tarkowska, A., Walczysko, P., Allan, C., Burel, J.-M., and Swedlow, J. (2015). OMERO and bio-formats 5: flexible access to large bioimaging datasets at scale. In Ourselin, S. and Styner, M. A., editors, *SPIE Proceedings*. SPIE.
- [Naderi et al., 2011] Naderi, H., Matin, M. M., and Bahrami, A. R. (2011). Review paper: Critical issues in tissue engineering: Biomaterials, cell sources, angiogenesis, and drug delivery systems. *Journal of Biomaterials Applications*, 26:383–417.
- [National Institute of Biomedical Imaging and Bioengineering, 2023] National Institute of Biomedical Imaging and Bioengineering (2023). Glossary of terms. <https://www.nibib.nih.gov/science-education/glossary/b>. Accessed March 20, 2023.
- [Nwaneshiudu et al., 2012] Nwaneshiudu, A., Kuschal, C., Sakamoto, F. H., Anderson, R. R., Schwarzenberger, K., and Young, R. C. (2012). Introduction to confocal microscopy. *Journal of Investigative Dermatology*, 132(12):1–5.
- [OpenCV, 2023] OpenCV (2023). *Miscellaneous Image Transformations*. OpenCV. Accessed April 29, 2023.
- [Pastore and Calcagnì, 2019] Pastore, M. and Calcagnì, A. (2019). Measuring distribution similarities between samples: A distribution-free overlapping index. *Frontiers in Psychology*, 10.
- [Preem and Kogermann, 2018] Preem, L. and Kogermann, K. (2018). Electrospun antimicrobial wound dressings: Novel strategies to fight against wound infections. In *Chronic Wounds, Wound Dressings and Wound Healing*, pages 213–253. Springer International Publishing.
- [Rafael C. Gonzalez, 2018] Rafael C. Gonzalez, R. E. W. (2018). *Digital Image Processing*. 4 edition.
- [Rezakhaniha et al., 2011] Rezakhaniha, R., Agianniotis, A., Schrauwen, J. T. C., Griffa, A., Sage, D., Bouten, C. V. C., van de Vosse, F. N., Unser, M., and Stergiopoulos, N. (2011). Experimental investigation of collagen waviness and orientation in the arterial adventitia using confocal laser scanning microscopy. *Biomechanics and Modeling in Mechanobiology*, 11(3-4):461–473.
- [Ronneberger et al., 2015] Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Lecture Notes in Computer Science*, pages 234–241. Springer International Publishing.

- [Salalha et al., 2006] Salalha, W., Kuhn, J., Dror, Y., and Zussman, E. (2006). Encapsulation of bacteria and viruses in electrospun nanofibres. *Nanotechnology*, 17(18):4675–4681.
- [Stanger et al., 2014] Stanger, J., Tucker, N., Buunk, N., and Truong, Y. (2014). A comparison of automated and manual techniques for measurement of electrospun fibre diameter. *Polymer Testing*, 40:4–12.
- [Tathe et al., 2010] Tathe, A. C., Ghodke, M. S., Nikalje, A. P. G., and Chavan, Y. B. (2010). A brief review: Biomaterials and their application.
- [ThermoFisher Scientific, 2023] ThermoFisher Scientific (2023). Live/dead™ bacLight™ bacterial viability kit, for microscopy quantitative assays. <https://www.thermofisher.com/order/catalog/product/L7012>. Accessed April 5, 2023.
- [Tomba et al., 2010] Tomba, E., Facco, P., Roso, M., Modesti, M., Bezzo, F., and Barolo, M. (2010). Artificial vision system for the automatic measurement of interfiber pore characteristics and fiber diameter distribution in nanofiber assemblies. *Industrial & Engineering Chemistry Research*, 49(6):2957–2968.
- [van Raaphorst et al., 2020] van Raaphorst, R., Kjos, M., and Veening, J. W. (2020). Bactmap: An r package for integrating, analyzing and visualizing bacterial microscopy data. *Molecular Microbiology*, 113:297–308.
- [Vernon-Parry, 2000] Vernon-Parry, K. (2000). Scanning electron microscopy: an introduction. *III-Vs Review*, 13(4):40–44.
- [Weng and Xie, 2015] Weng, L. and Xie, J. (2015). Smart electrospun nanofibers for controlled drug release: Recent advances and new perspectives. *Current Pharmaceutical Design*, 21(15):1944–1959.
- [Xie et al., 2008] Xie, X. S., Choi, P. J., Li, G. W., Nam, K. L., and Lia, G. (2008). Single-molecule approach to molecular biology in living bacterial cells.
- [Yao et al., 2014] Yao, J., Bastiaansen, C., and Peijs, T. (2014). High strength and high modulus electrospun nanofibers. *Fibers*, 2(2):158–186.
- [Zeiss, 2023] Zeiss (2023). Confocal microscopes. <https://www.zeiss.com/microscopy/en/products/light-microscopes/confocal-microscopes.html>. Accessed April 9, 2023.
- [Zupančič et al., 2019] Zupančič, Š., Škrlec, K., Kocbek, P., Kristl, J., and Berlec, A. (2019). Effects of electrospinning on the viability of ten species of lactic acid bacteria in poly(ethylene oxide) nanofibers. *Pharmaceutics*, 11(9):483.

[Zussman, 2010] Zussman, E. (2010). Encapsulation of cells within electrospun fibers. *Polymers for Advanced Technologies*, 22(3):366–371.

Appendix

I. U-Net architecture

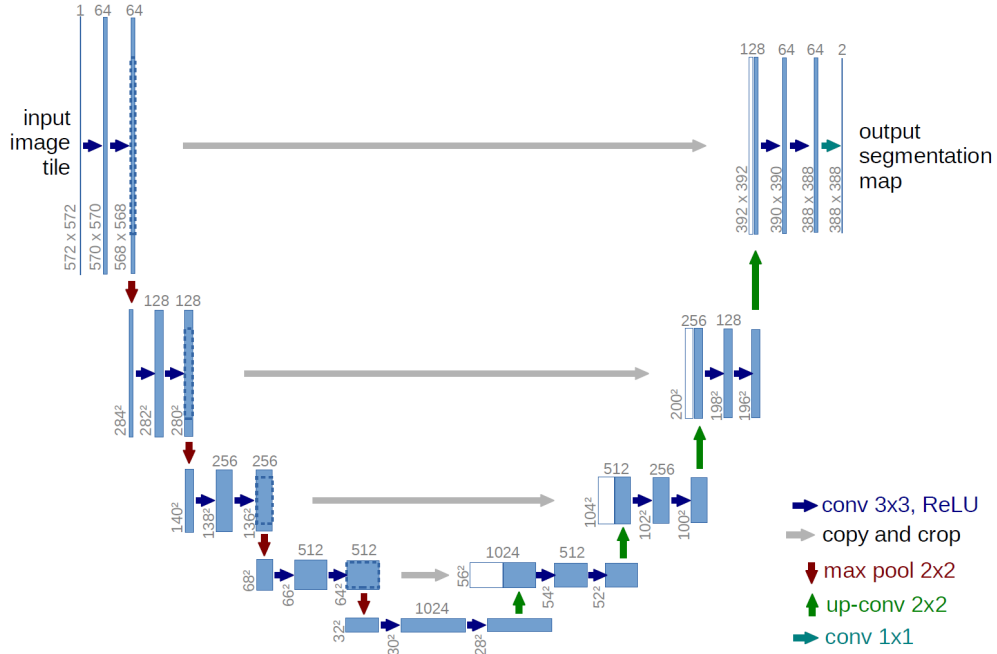


Figure A1. Example U-Net architecture from the original paper [Ronneberger et al., 2015]. Each blue box is a multi-channel feature map where the number of channels is shown on top of the box. The height and width are provided at the lower left edge of the box. White boxes represent copied feature maps and the arrows denote the different operations as can be seen from the legend on the right. The shape of the input image in this thesis was 256x256x1 and the shape was 16x16 in the lowest resolution. Dropout of 0.1 was also applied to most of the convolutional layers in the network in this thesis.

II. Link to the project repository

The link to the Github code repository is: https://github.com/marilin99/master_thesis. The "dev" branch is for development, while the "main" branch is for users who would like to setup the tool or test the tool out. The instructions on how to test the pipelines out quickly is provided in the "README.md" of the main branch.

III. Licence

Non-exclusive licence to reproduce thesis and make thesis public

I, **Marilyn Moor**,
(author's name)

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright,

"FiBar": a Tool for Automated Analysis of Complex Biomaterials from Microscopy Images,
(title of thesis)

supervised by Dmytro Fishman, Marta Putrinš and Karin Kogermann.
(supervisor's name)

2. I grant the University of Tartu a permit to make the work specified in p. 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons licence CC BY NC ND 3.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright.
3. I am aware of the fact that the author retains the rights specified in p. 1 and 2.
4. I certify that granting the non-exclusive licence does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Marilyn Moor
09/05/2023