

Daniel Hein · M.-Nr. 03627602

## 1 Data Preprocessing

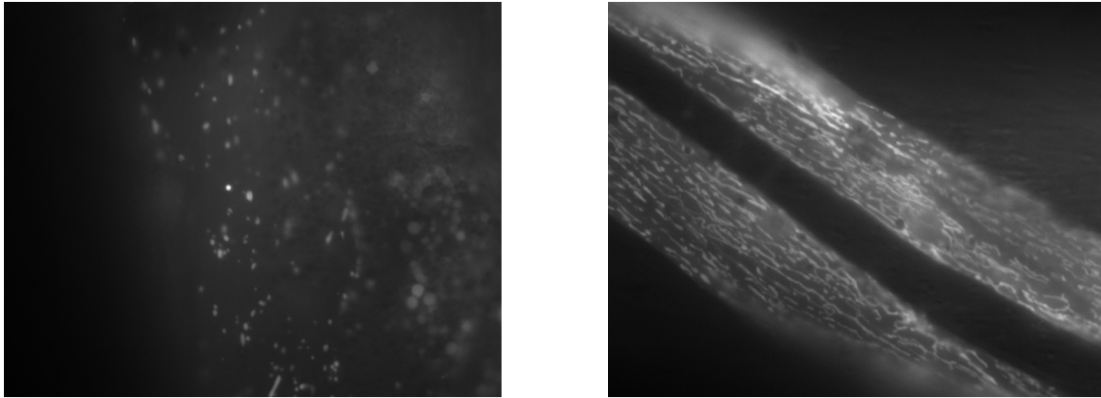
To extract relevant features from the images preprocessing is necessary. To do so we created a preprocessing pipeline that consists of five separate functions.

### Sharpest image.

INPUT: grayscale image matrix  $I$

OUTPUT: estimated sharpness  $s$

VARIABLES TO ADJUST: -



Each image stack consists between around 8 and 18 photography's of the same part of one worm. The images differ only in the focal plane that is shown. In the most cases only one of these images shows the mitochondria in a sharp and clear way that can be used to classify the mitochondria. The manual process is to look at each of the images, select the sharpest and proceed with the next classification tasks only with this one. If a program should become able to classify automatically, it is necessary to find a way to select the sharpest image automatically, too.

For this purpose a simple gradient based estimation function is used. The grayscale image is represented by the matrix  $I \subseteq \mathbb{N}^{m \times n}$  with values in the range  $[0, 255]$  with 0 for black and 255 for white pixels. The sum over the gradients  $I_x$  and  $I_y$  in both image dimensions  $x$  and  $y$  normalized by the number of gradients gives a good estimation on how sharp a photography actually is.

$$I_x = \frac{\partial I}{\partial x} \quad I_y = \frac{\partial I}{\partial y} \quad (1)$$

$$s = \frac{\sum_{i=1}^m \sum_{j=1}^n S_{ij}}{mn} \quad (2)$$

$$S_{ij} = \sqrt{(I_{xij})^2 + (I_{yij})^2} \quad (3)$$

The higher the value  $s$  the sharper the image. A disadvantage of this simple method is that one usually can only compare images showing the same motive and so in real image processing software much better methods for estimating image sharpness are in use. Nevertheless it has shown to be a sufficient

method for our purposes. Because we are only comparing images within on single stack and they all show the exact same microscope photography.

### Reduced noise.

INPUT: grayscale image matrix  $I$

OUTPUT: grayscale image matrix  $I$

VARIABLES TO ADJUST: neighborhoods of size  $m$ -by- $n$  to estimate the local image mean and standard deviation

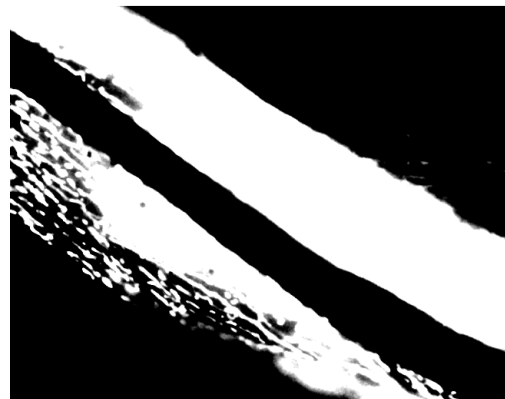
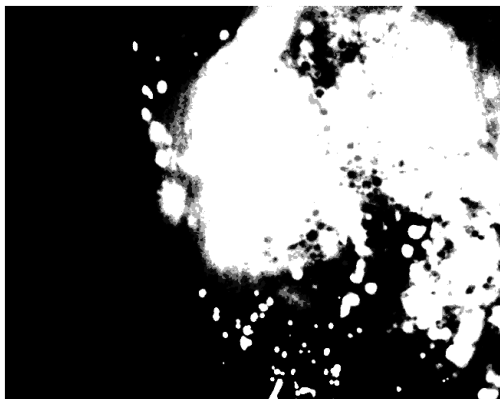
Due to the microscope environment the pictures are noisy. One reason is for example the sensor of the build-in camera. Reducing the noise will improve finding mitochondria in the images. We decided to use a pixelwise adaptive Wiener method based on statistics estimated from a local neighborhood of each pixel. For this reason the value 3 for  $m$  and  $n$  has proven convenient.

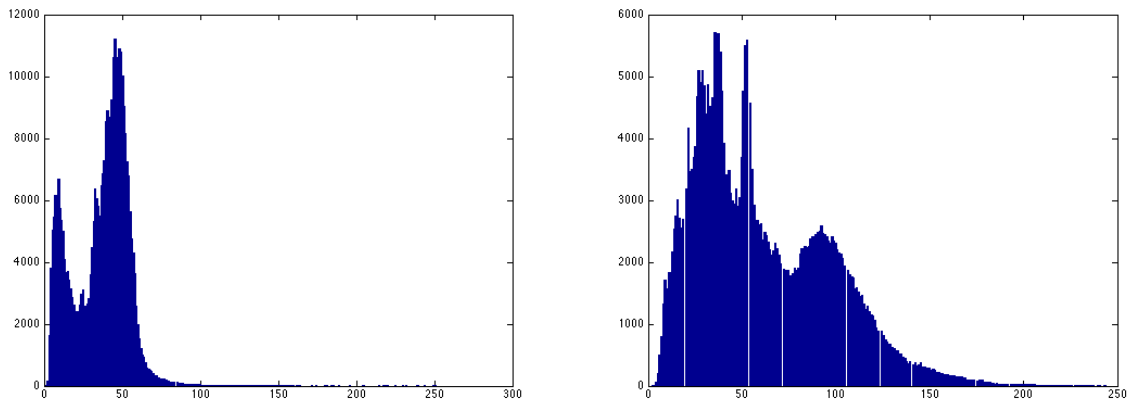
### Histogram and image contrast.

INPUT: grayscale image matrix  $I$

OUTPUT: grayscale image matrix  $I$

VARIABLES TO ADJUST:  $\alpha$  and  $\beta$  representing percentages of how many pixels in  $I$  become black and how many become white





To adjust the optimal contrast of each image it is necessary to have a look on the histograms and how the different levels (0 to 255) of gray are distributed all over one image. The task is to find suitable values for  $b_l$  the lower bound below which everything goes to 0 (black) and the  $b_u$  the upper bound above which everything goes to 255 (white).

$$f_l(l) = \sum_{i=1}^l h_i \leq \beta \sum_{i=1}^{256} h_i \quad f_u(u) = \sum_{i=1}^u h_{256-i+1} \leq \alpha \sum_{i=1}^{256} h_i \quad (4)$$

$$1 \leq l \leq 256 \quad 1 \leq u \leq 256 \quad (5)$$

$$b_l = \frac{\operatorname{argmax}_b(f_B)}{256} \quad b_u = \frac{\operatorname{argmax}_u(f_u)}{256} \quad (6)$$

$$\alpha + \beta \leq 1 \quad (7)$$

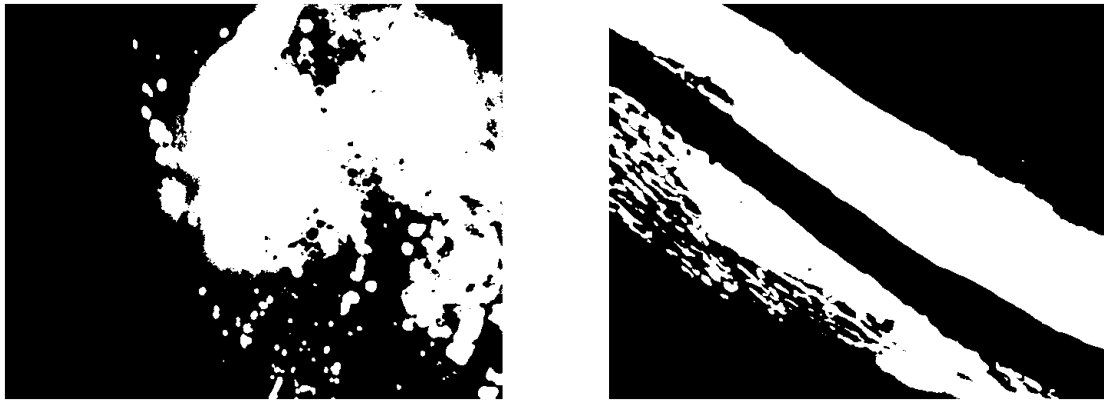
The gray scale levels in between are scaled to fit into the histogram of the output image. The result is a high contrast image in which mitochondria are much easier to find. Because of all the images have different histograms it is not possible to use the same lower bound and upper bound for all of them. So we defined the new variables black percentage and white percentage corresponding to how many of the pixels should become 0 (black) or 255 (white) after the adjustment. Using this procedure we are able to find the optimal bounds for each individual image. The performance of the later classification process will highly depend on the output quality of this contrast adjustment. In order to find the best values for the variables black percentage and white percentage we used the optimization tools of MatLab. The variables to optimize for are black percentage and white percentage and the error to minimize is the misclassification rate on the training set.

### Binary image.

INPUT: grayscale image matrix  $I$

OUTPUT: binary image matrix  $I_{bw}$

VARIABLES TO ADJUST: level above which all pixels become white and below which all pixels become black



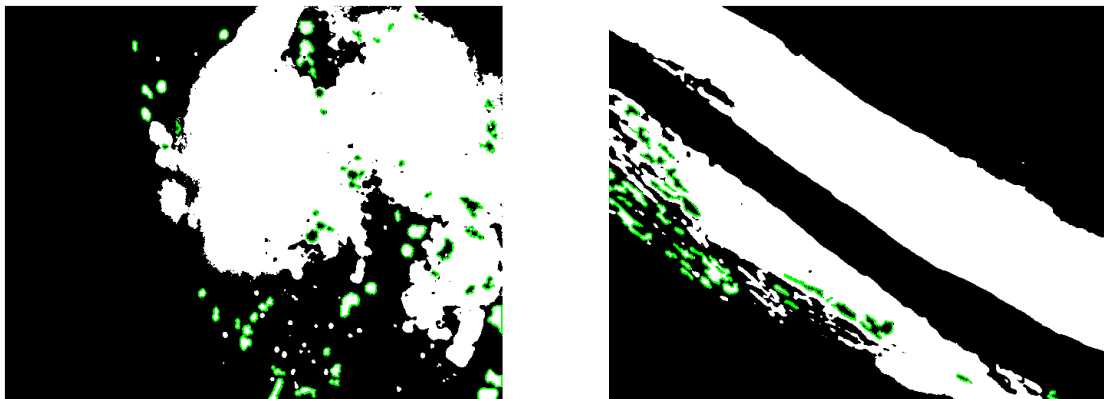
After improving the contrast of each image it can easily be converted to a binary image in that the mitochondria can be found as white areas.

### Boundaries.

INPUT: binary image matrix  $I_{bw}$

OUTPUT: vector  $b$  containing mitochondria boundaries

VARIABLES TO ADJUST: values of pixels below and above that a boundary is not describing valid mitochondria



Using a standard MatLab function it is possible to trace the exterior boundaries of the mitochondria in the binary image. After filtering out objects that are too big or too small to be valid mitochondria we have generated a set of vectors each containing the borders of one of the mitochondria.