

Supplementary Information for:

Vimentin intermediate filaments structure and mechanically support microtubules in cells

Anna Blob,[†] David Ventzke,^{†,‡} Giacomo Nies,[‡] Jan Niklas Dühmert,[¶] Bernhard Schmitzer,[¶] Axel Munk,^{‡,§} Laura Schaedel,^{||} and Sarah Köster^{*,†,§}

[†]*Institute for X-Ray Physics, University of Göttingen, Friedrich-Hund-Platz 1, 37077 Göttingen, Germany*

[‡]*Institute for Mathematical Stochastics, University of Göttingen, Goldschmidtstraße 7, 37077 Göttingen, Germany*

[¶]*Institute of Computer Science, University of Göttingen, Goldschmidtstraße 7, 37077 Göttingen, Germany*

[§]*Cluster of Excellence "Multiscale Bioimaging: from Molecular Machines to Networks of Excitable Cells" (MBExC), University of Göttingen, Germany*

^{||}*Department of Physics, Center for Biophysics, Saarland University, Campus A2 4, 66123 Saarbrücken, Germany*

E-mail: sarah.koester@uni-goettingen.de

Supplementary materials and methods

Image preprocessing

Every image of microtubules and vimentin IFs is processed following the workflow shown in Fig. 1 to ensure the same position and orientation of the cell. First, images of crossbows are rotated by 180° , if necessary. Second, all cell images are spatially shifted based on a segmentation of the interior of the cell: Each microtubule image is uniquely background corrected by subtracting the minimum intensity value. A Gaussian filter is applied and the image is binarized by thresholding with Otsu's method. Any small holes remaining in the segmented cell area are filled and a binary erosion is applied. The resulting binary image is shifted to position the center of mass at the same pixel for all images. This aligned, segmented cell image I_{segm} is used as an auxiliary image for further analysis steps. The same shift that is used for the segmented image is applied to the correctly oriented fluorescence image to obtain an aligned fluorescence image as the starting point for all further analysis.

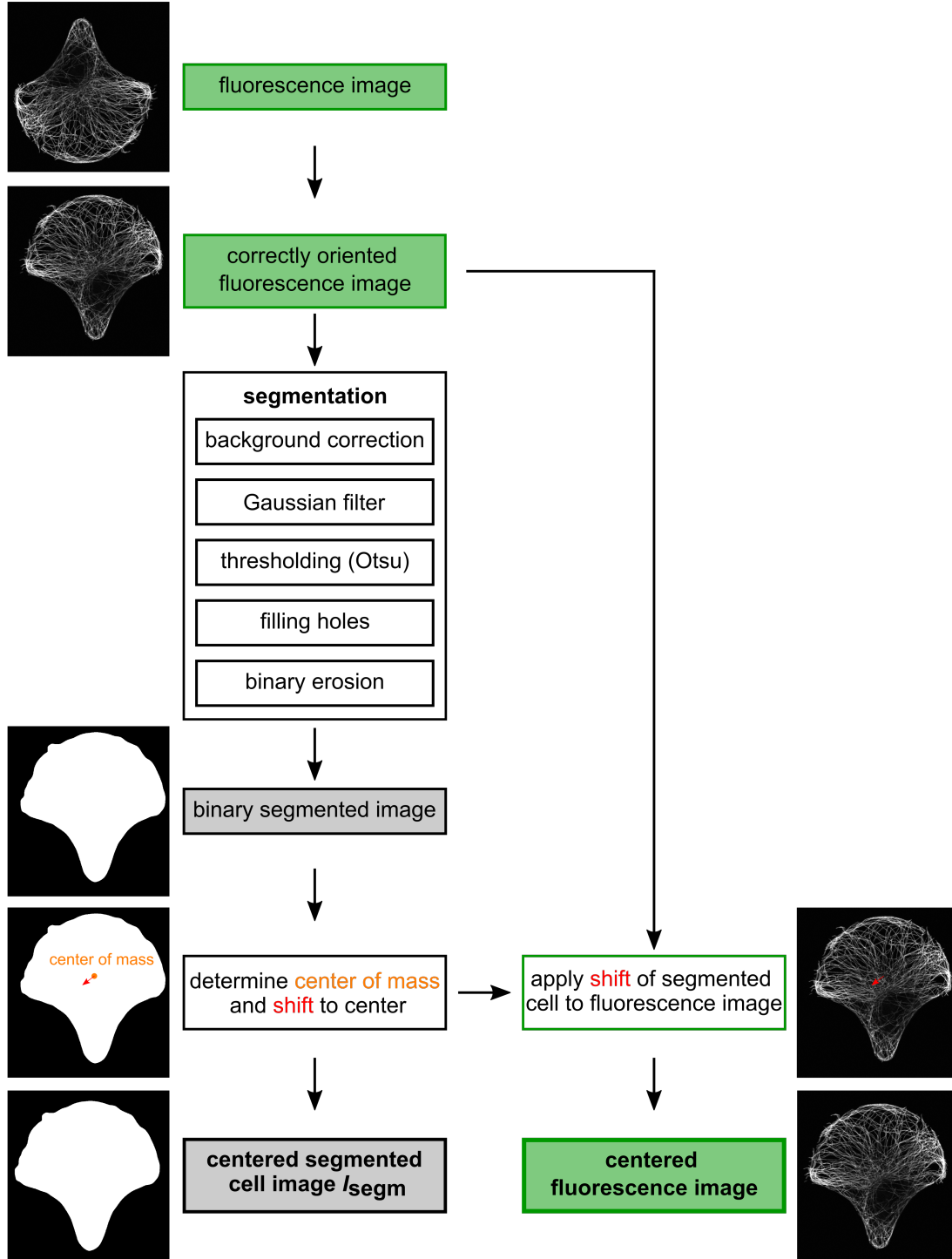


Figure 1: Workflow for preprocessing of all fluorescence images. All cells are aligned with respect to each other in terms of orientation and position. The aligned image I_{segm} is used as an auxiliary image for further analysis steps.

Binary cell area mask

A binary mask for the typical area of NIH3T3 cells is created by overlaying all segmented cell images of NIH3T3 cells of the same shape as schematically illustrated in Fig. 2. To exclude outliers of unusually large cell shapes only pixels belonging to more than 10% of all the segmented cells constitute together the binary cell mask.

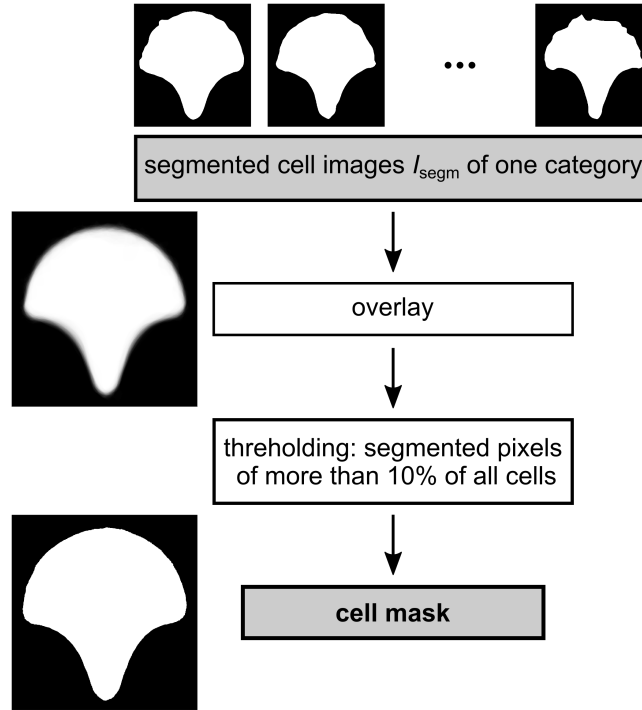


Figure 2: Workflow of processing steps to create a binary mask for cells of one category.

Microtubule skeletonization and filament extraction

The coordinates of the microtubules are extracted following a skeletonization procedure that is adopted from CytoSeg2.0¹ and represented in Fig. 3. A “tubeness filter” is applied to the aligned normalized fluorescence image of microtubules by computing the Hessian matrix of the intensity at each pixel and taking the negative of the smallest eigenvalue. The manually chosen parameter v_{width} is required as input for a measure of the width of the filaments. The tubeness-filtered image is binarized with an adaptive threshold based on a Gaussian weighted mean of the surrounding of each pixel. The standard deviation of the Gaussian for this threshold is determined by the second required input parameter v_{thresh} . The binarized image is then skeletonized² and a cell mask defining the cell area is applied to set all pixels outside of the region of interest to zero. The skeletonized image is further “cleaned” by removing skeleton components for which the mean intensity in the fluorescence image is lower than v_{int} times the mean intensity of the entire cell and removing skeleton components that are shorter than the threshold length v_{size} . All cells are processed with the same four input parameters v_{width} , v_{thresh} , v_{int} and v_{size} chosen such that after visual inspection the best possible skeletonization is achieved.

In the skeletonized microtubule networks, intersections (“nodes”) of microtubules and end-points of microtubules are defined as skeleton pixels with more than two neighboring skeleton pixels or only one neighboring skeleton pixel, respectively (“neighboring” refers to horizontal, vertical or diagonal connections). All other skeletonized pixels with exactly two neighbors are regarded as filament pixels. Filament pixels connected directly to each other form one filament together with the start and end node/endpoint. The pixel coordinates of such a filament are extracted by an algorithm that starts at one node or endpoint pixel, searches the next neighboring filament pixel, continues from this filament pixel to search the next one and repeats stepping forward until a node or endpoint is reached. The process is repeated until all filament pixels have been attributed to one filament. Note that one *filament* resulting from this extraction process corresponds to one segment of a real microtubule only,

since the filament tracing does not continue across nodes. For simplicity, we still refer to these extracted filament segments as filaments. For the subsequent curvature analysis only filaments that are longer than 20 pixels are taken into account and for each filament the outer 3 pixels on each side are omitted to reduce the effect of skeletonization artifacts.

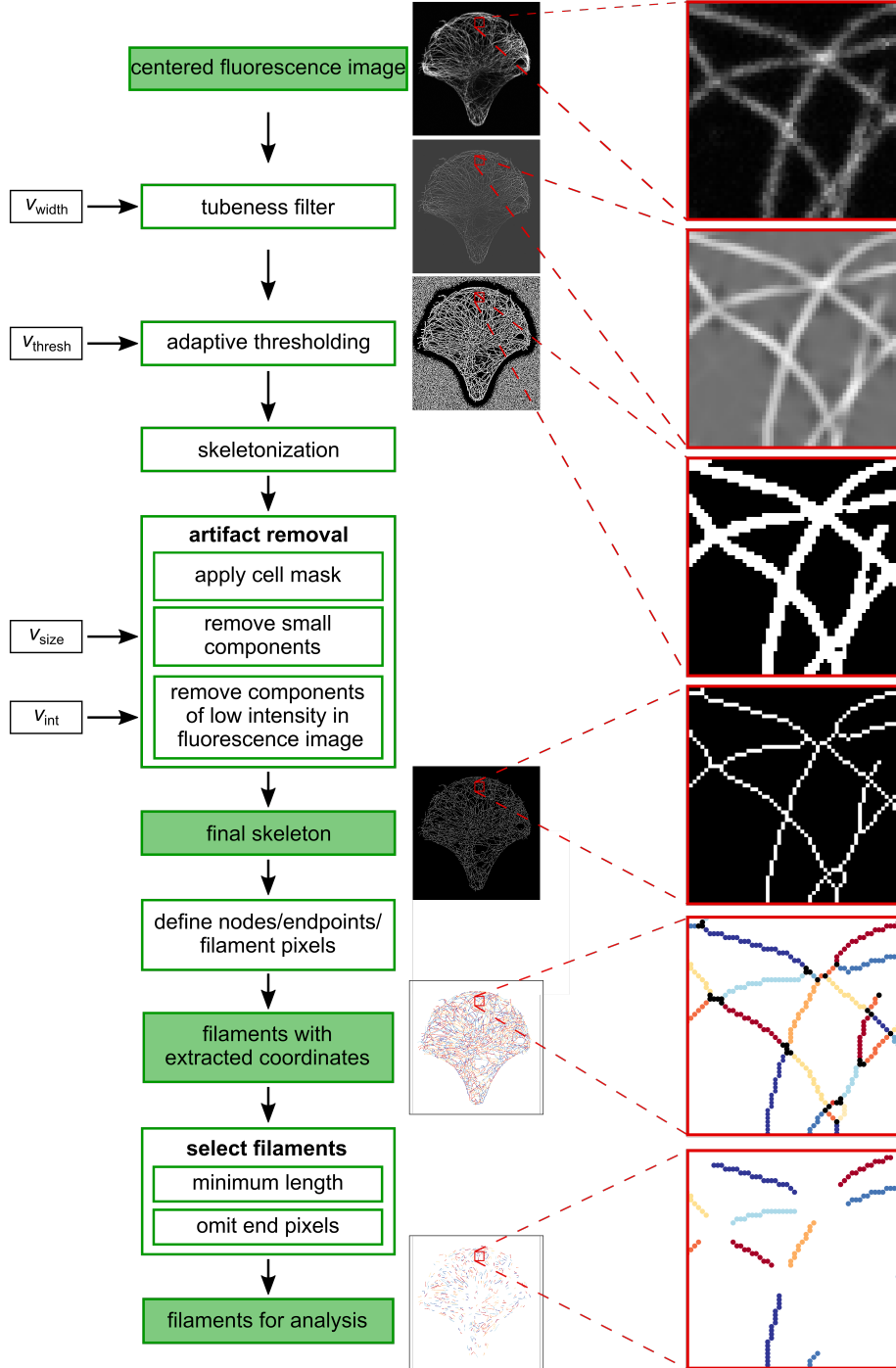


Figure 3: Workflow of the skeletonization and filament extraction procedure from microtubule fluorescence images. The four manually chosen parameters v_{width} , v_{thresh} , v_{int} and v_{size} are required as input for the skeletonization algorithm. From the skeletonized microtubules, the coordinates of the segments between nodes are extracted. The colors for the extracted filaments serve only to differentiate between different segments. The node pixels are represented in black and can belong to several filaments at the same time. For the subsequent curvature analysis only filaments longer than 20 pixels are considered and their outer 3 pixels on each side are omitted.

Principal component analysis of curvature histograms

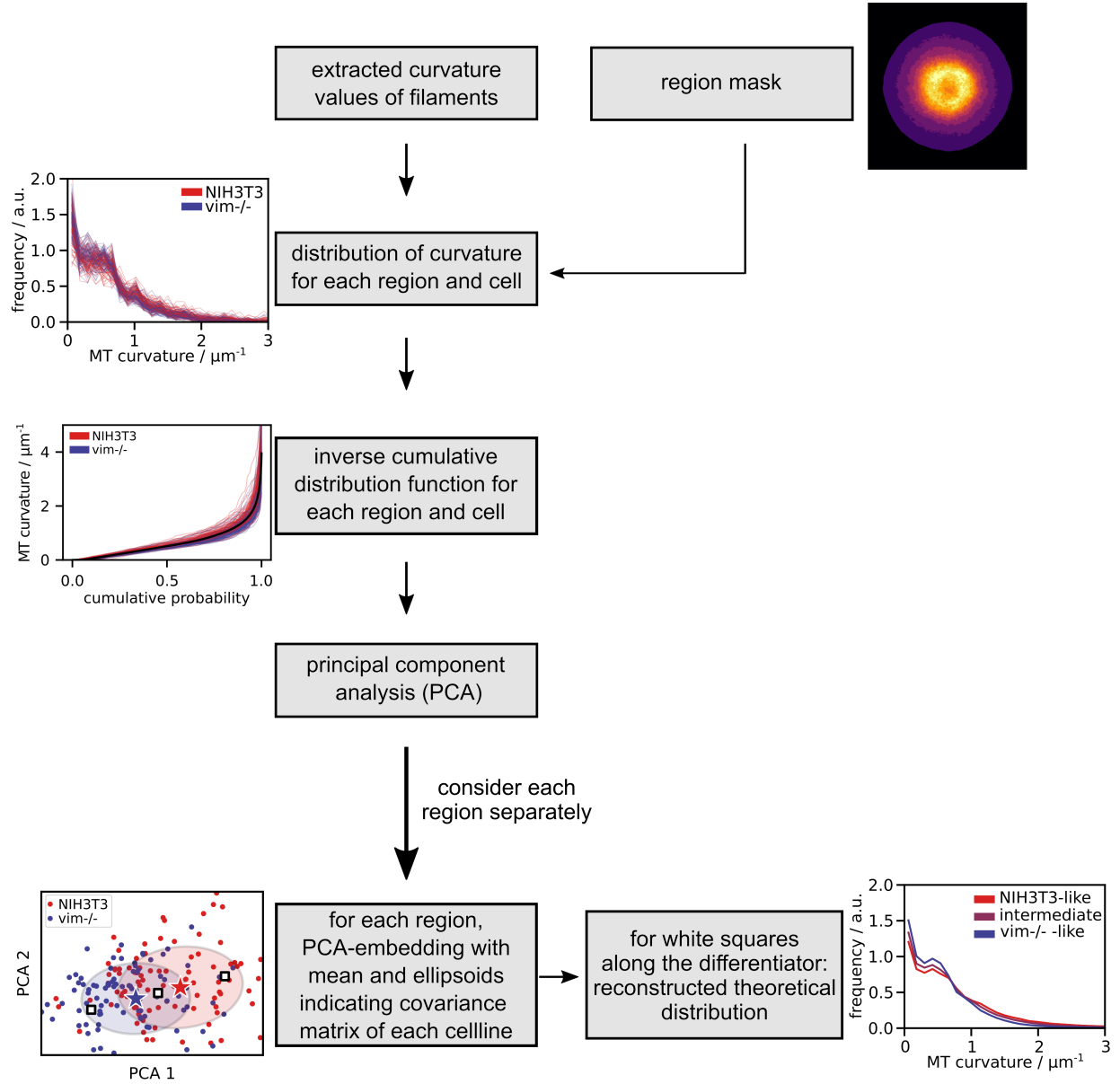


Figure 4: Workflow for the principal component analysis of curvature histograms.

Calculation of average microtubule curvature in dependence of vimentin IF density and corresponding confidence bands

Let $n \in \mathbb{N}$ be the total number of pixels, for which we perform the curvature analysis in the cell images, and $g \in \mathbb{N}$ the total number of the detected filaments. (In our case, we have that $n = 638\,427$ and $g = 34\,518$ for the crossbows and $n = 629\,641$ and $g = 35\,820$ for the circles.) For every pixel we measure the curvature value y , the vimentin intensity x and its length v . Hence, our dataset for this analysis consists of n triplets of values $(x_1, y_1, v_1), \dots, (x_n, y_n, v_n)$.

To study the relation between the curvature and the vimentin intensity, we consider a weighted rolling average function (conditional kernel regression estimator) $\hat{m} : \mathbb{R} \rightarrow \mathbb{R}$ given by,

$$\hat{m}(x) = \frac{\sum_{i=1}^n y_i v_i k(x_i, x)}{\sum_{i=1}^n v_i k(x_i, x)} \quad (1)$$

where we used a normal kernel

$$k(x_i, x) = \frac{1}{h\sqrt{2\pi}} e^{\frac{1}{2}\left(\frac{x_i - x}{h}\right)^2}$$

and a standard deviation of $h = 0.1$. For a fixed $x \in \mathbb{R}$ this statistic of the data constitutes an approximation of the average curvature of a segment of a filament, conditioned on its vimentin intensity being equal to x .

In order to quantify the variability of the estimated curvature, we also provide pointwise confidence bands for the population counterpart of $\hat{m}(x)$, which can be thought of as the limit value we would obtain if we had access to data from an infinite number of cells. Even though the finite sample distribution of $\hat{m}(x)$ is unknown, we can determine the confidence interval using a cluster bootstrap procedure³. This procedure consists in sampling $g = 35\,820$ times for circles and $g = 34\,518$ for the crossbows with replacement from the detected filaments (clusters) $F_1, \dots, F_g$, instead of sampling from the pixels. We then construct our bootstrapped datasets by considering the curvature, vimentin intensity and weight of the

pixels that constitute the sampled filaments. This clustered resampling method has the advantage of taking into account the strong correlation between curvature values of pixels that belong to the same filament. After constructing $B = 1\,000$ resampled datasets, we compute the corresponding bootstrap replicas $\hat{m}^{(j)}$, with $j = 1, \dots, B$, as in Eq. (1). For each, $x \in \mathbb{R}$ we can then determine a basic bootstrap confidence interval $I(x) \subset \mathbb{R}$, also known as the reverse percentile interval, given by:

$$I(x) = (2\hat{m}(x) - m_{(1-\alpha/2)}^*(x), 2\hat{m}(x) - m_{(\alpha/2)}^*(x))$$

where $m_{(1-\alpha/2)}^*(x)$ denotes the $1-\alpha/2$ percentile of the bootstrapped replicas $\hat{m}^{(1)}(x), \dots, \hat{m}^{(1000)}$. We set $\alpha = 0.05$ to calculate a 95% pointwise confidence band.

References

- (1) Breuer, D.; Nowak, J.; Ivakov, A.; Somssich, M.; Persson, S.; Nikoloski, Z. System-wide organization of actin cytoskeleton determines organelle transport in hypocotyl plant cells. *Proc. Natl. Acad. Sci. USA* **2017**, *114*, E5741–E5749.
- (2) Lee, T.; Kashyap, R.; Chu, C. Building Skeleton Models via 3-D Medial Surface Axis Thinning Algorithms. *CVGIP: Graphical Models and Image Processing* **1994**, *56*, 462–478.
- (3) Cameron, A. C.; Gelbach, J. B.; Miller, D. L. Bootstrap-Based Improvements for Inference with Clustered Errors. *Rev. Econ. & Stat.* **2008**, *90*, 414–427.