Supporting Information for

FiberApp: an Open-source Software for Tracking and Analyzing Biomacromolecules, Polymers, Filaments and Fibrous Objects

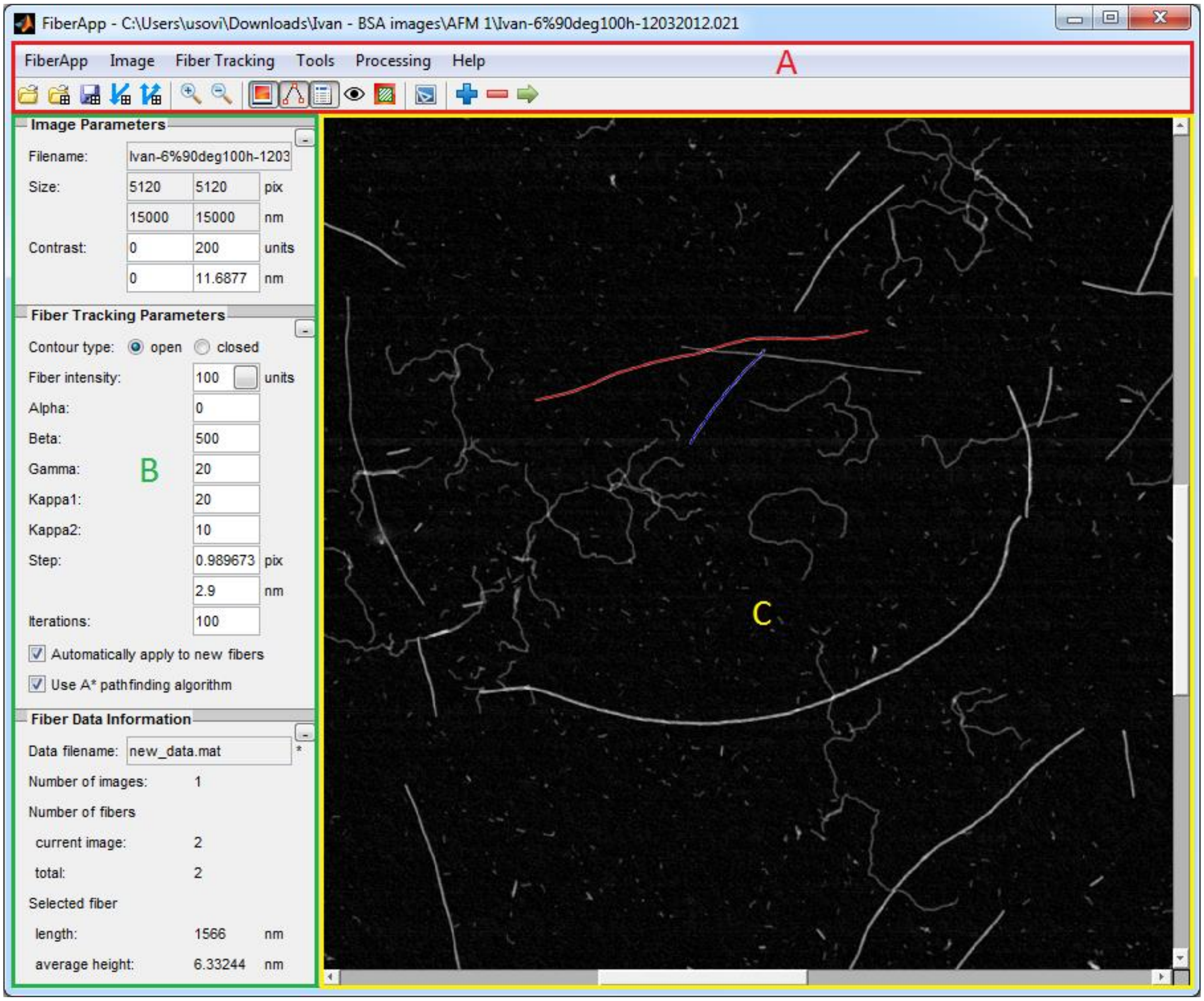
Ivan Usov and Raffaele Mezzenga

ETH Zurich, Department of Health Science & Technology, Schmelzbergstrasse 9, LFO E23, 8092 Zurich

FiberApp is a compiled MATLAB application and it can be launched on any computer with MATLAB Compiler Runtime 8.1 (MCR) being installed. The program is an open-source software and the source code can be viewed and downloaded at the website www.fsm.ethz.ch

A brief introduction in a working process with FiberApp can be activated *via* **Help → Tutorial**.

The *FiberApp* main window has three main areas, as it is shown in Figure S1. The area (A) covers the main menu and toolbars with commonly used action buttons. Most of the functionality is accessible *via* special panels that are situated on the left side of the window, and the area (B) highlights panels that are currently open and active. Finally, image viewing and tracking of fiberlike objects can be performed in the active area (C).



FiberApp - C:\Users\usovi\Downloads\Ivan - BSA images\AFM 1\Ivan-6%90deg100h-12032012.021

Image Parameters

Filename: Ivan-6%90deg100h-1203

Size: 5120 5120 pix

15000 15000 nm

Contrast: 0 200 units

0 11.6877 nm

Fiber Tracking Parameters

Contour type: ● open ○ closed

Fiber intensity: 100 units

Alpha: 0

Beta: 500

Gamma: 20

Kappa1: 20

Kappa2: 10

Step: 0.989673 pix

2.9 nm

Iterations: 100

☑ Automatically apply to new fibers

☑ Use A* pathfinding algorithm

Fiber Data Information

Data filename: new_data.mat *

Number of images: 1

Number of fibers

current image: 2

total: 2

Selected fiber

length: 1566 nm

average height: 6.33244 nm

Figure S1. The main window of the *FiberApp* open-source software. (A) The program's main menu and toolbars with commonly used action buttons. (B) *FiberApp* functional panels for accessing fiber properties, information and data analysis tools. (C) Active area for image navigation and fiber tracking.

A broad set of functional tools of the program is represented in Figure S2. There are 5 core panels for image, mask and fiber tracking parameters, as well as tracked data information and fiber view properties. One panel is for generating images with simulated fibrils and corresponding XYZ data, which serves the purpose of tracking quality and algorithm correctness validation. In the current version there are 14 data processing tools, allowing determination of the basic single-fiber morphological parameters, distributions, collective orientation behavior, etc.

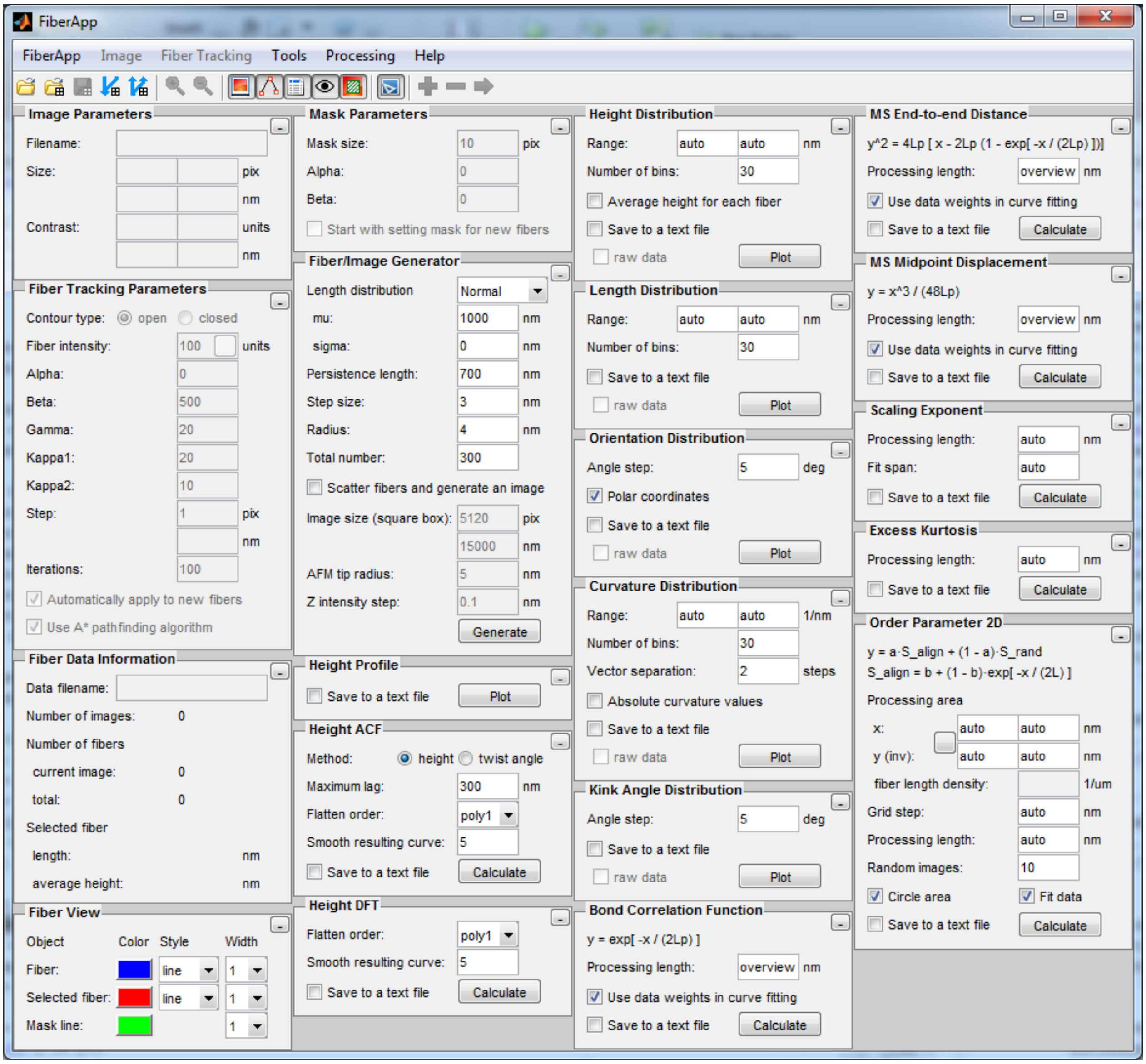


Image Parameters

Filename:

Size: pix

nm

Contrast: units

nm

Fiber Tracking Parameters

Contour type: ○ open ○ closed

Fiber intensity: 100 units

Alpha: 0

Beta: 500

Gamma: 20

Kappa1: 20

Kappa2: 10

Step: 1 pix

nm

Iterations: 100

☑ Automatically apply to new fibers

☑ Use A* pathfinding algorithm

Fiber Data Information

Data filename:

Number of images: 0

Number of fibers

current image: 0

total: 0

Selected fiber

length: nm

average height: nm

Fiber View

Object	Color	Style	Width
Fiber:	🟦	line	1
Selected fiber:	🟥	line	1
Mask line:	🟩		1

Mask Parameters

Mask size: 10 pix

Alpha: 0

Beta: 0

□ Start with setting mask for new fibers

Fiber/Image Generator

Length distribution Normal

mu: 1000 nm

sigma: 0 nm

Persistence length: 700 nm

Step size: 3 nm

Radius: 4 nm

Total number: 300

□ Scatter fibers and generate an image

Image size (square box): 5120 pix

15000 nm

AFM tip radius: 5 nm

Z intensity step: 0.1 nm

Generate

Height Profile

□ Save to a text file Plot

Height ACF

Method: ● height ○ twist angle

Maximum lag: 300 nm

Flatten order: poly1

Smooth resulting curve: 5

□ Save to a text file Calculate

Height DFT

Flatten order: poly1

Smooth resulting curve: 5

□ Save to a text file Calculate

Height Distribution

Range: auto auto nm

Number of bins: 30

□ Average height for each fiber

□ Save to a text file

□ raw data Plot

Length Distribution

Range: auto auto nm

Number of bins: 30

□ Save to a text file

□ raw data Plot

Orientation Distribution

Angle step: 5 deg

☑ Polar coordinates

□ Save to a text file

□ raw data Plot

Curvature Distribution

Range: auto auto 1/nm

Number of bins: 30

Vector separation: 2 steps

□ Absolute curvature values

□ Save to a text file

□ raw data Plot

Kink Angle Distribution

Angle step: 5 deg

□ Save to a text file

□ raw data Plot

Bond Correlation Function

y = exp[-x / (2Lp)]

Processing length: overview nm

☑ Use data weights in curve fitting

□ Save to a text file Calculate

MS End-to-end Distance

y^2 = 4Lp [x - 2Lp (1 - exp[-x / (2Lp)])]

Processing length: overview nm

☑ Use data weights in curve fitting

□ Save to a text file Calculate

MS Midpoint Displacement

y = x^3 / (48Lp)

Processing length: overview nm

☑ Use data weights in curve fitting

□ Save to a text file Calculate

Scaling Exponent

Processing length: auto nm

Fit span: auto

□ Save to a text file Calculate

Excess Kurtosis

Processing length: auto nm

□ Save to a text file Calculate

Order Parameter 2D

y = a·S_align + (1 - a)·S_rand

S_align = b + (1 - b)·exp[-x / (2L)]

Processing area

x: auto auto nm

y (inv): auto auto nm

fiber length density: 1/um

Grid step: auto nm

Processing length: auto nm

Random images: 10

☑ Circle area ☑ Fit data

□ Save to a text file Calculate

Figure S2. The main window with all functional panels active. Within them are 5 core panels: Image Parameters, Fiber Tracking Parameters, Fiber Data Information, Fiber View, and Mask Parameters; one panel for the fiber data (WLC model) generation and respective AFM image simulation: Fiber/Image Generator and, lastly, 14 panels of fiber data processing tools: Height Profile, Height Autocorrelation function (ACF), Height Discrete Fourier Transform (DFT), Height Distribution, Length Distribution, Orientation Distribution, Curvature Distribution, Kink Angle Distribution, Bond Correlation Function (BCF), Mean-Squared End-to-end Distance (MSED), Mean-Squared Midpoint Displacement (MSMD), Scaling Exponent, Excess Kurtosis, and Order Parameter 2D.

Initial image pre-processing

Image opening and its properties

To open a new image select **FiberApp** → **Open Image** or click on the corresponding icon in the toolbar. 📂 Possible image formats are NanoScope (v8.10) AFM image files and *.tif files for other types of microscopy images. In the latter case, a user will be prompted to specify a scale bar position and its length in nanometers (Figure S3) in order to acquire scaled results from data analysis by selecting **Image** → **Scale Bar**.

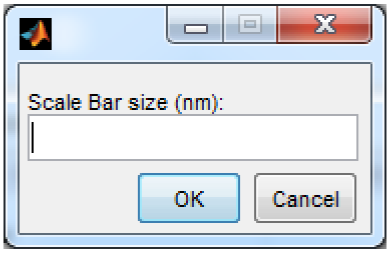
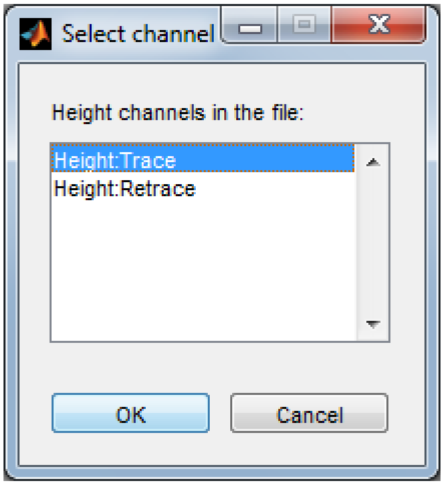


Figure S3. Dialog for the image width input upon loading *tif image files.

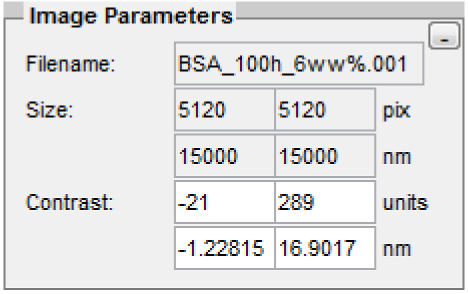
For NanoScope files with several height channels (e.g. trace, retrace), a user needs to choose one of them to work with, as it is shown in Figure S4.



Height:Trace
Height:Retrace

Figure S4. Dialog for the height channel selection from the AFM image.

It is required for the tracking procedure that fiber-like objects have higher pixel values (intensity) than the background. In the opposite case, inversion of the image is needed and can be accomplished by selecting **Image** → **Invert** in the menu.



Filename:	BSA_100h_6ww%.001		
Size:	5120	5120	pix
	15000	15000	nm
Contrast:	-21	289	units
	-1.22815	16.9017	nm

Figure S5. Image Parameters panel.
Filename – name of the image file on the disk (not editable).
Size – image size in pixels (top fields) and nanometers (bottom fields) (not editable).
Contrast – visual z-range of the image. The whole colormap spans within specified values in pixel's intensity units (top fields) and nanometers (bottom fields).

Images in TIF format do not contain the information about the relative height of objects in real units (for instance, nanometers), thus, it is not defined how the pixel intensity corresponds to its height. All the information about the height of objects for such images is kept in arbitrary intensity units.

Properties of the opened image are displayed in the Image Parameters panel (Figure S5). This panel can be accessed *via* **FiberApp → Main Panels → Image Parameters** item, or *via* the toolbar toggle button .

Remove Surface and Set Zero Level

The polynomial flattening in NanoScope software is a common pre-procedure to correct a bowed background of raw AFM images. However, the influence of the physical surface tilt and non-linearity of a piezoelectric scanner cannot be precisely described as a polynomial surface and an unavoidable uneven illumination persists in the image (Figure S6 (A)). Hence, to correct the uneven illumination of the image and obtain absolute image Z-data (height) with high accuracy, we include an additional step – the top-hat flattening algorithm. The result from the top-hat filter is the image minus the morphological opening of that image: $I^* = I - I_{\text{open}}$, where the opening of the image is a morphological operation of dilation followed by a morphological operation of erosion, and both operations are carried out by a structural element S: $I_{\text{open}} = I \circ S = (I \ominus S) \oplus S$.[1] In other words, the opening estimates the background of an image. To perform the described action, select **Image → Remove Surface**. The top-hat filter sets a zero value to the lowest point of the resulting image and this can differ from the real surface level (Figure S6 (B)). Using **Image → Set Zero Level**, one can define a rectangle covering a background plain area without objects. The mean intensity value will be subtracted from the whole image, thus the zero level will be assigned to the average height in this rectangular box (Figure S6 (C)).

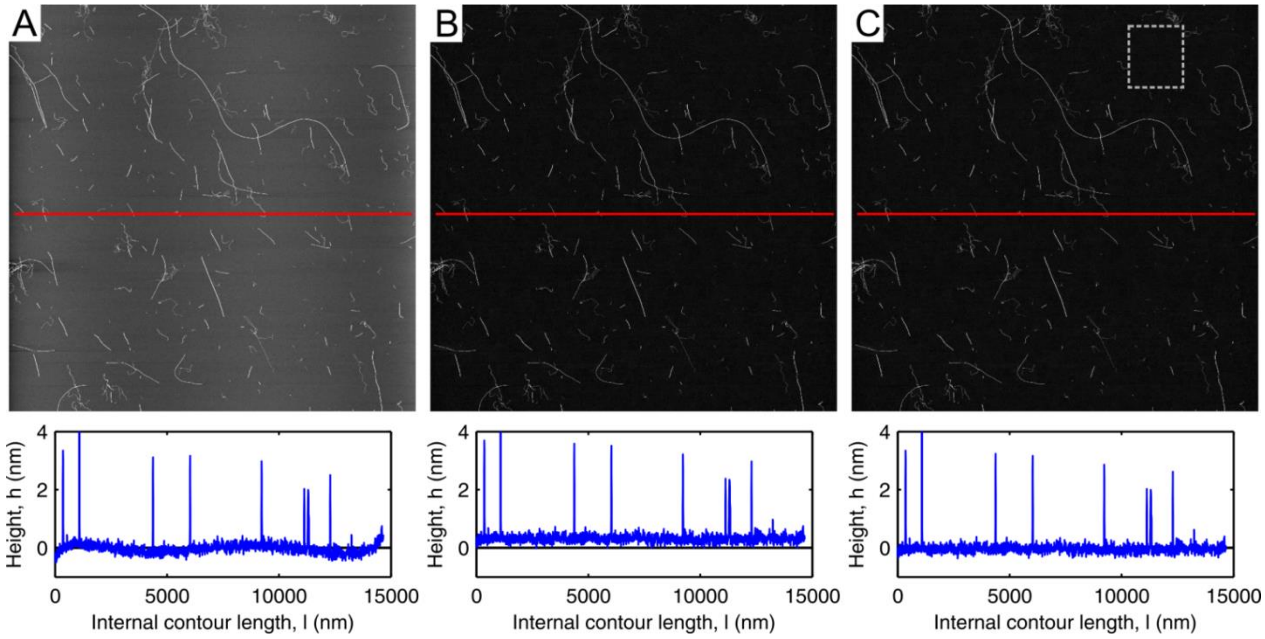
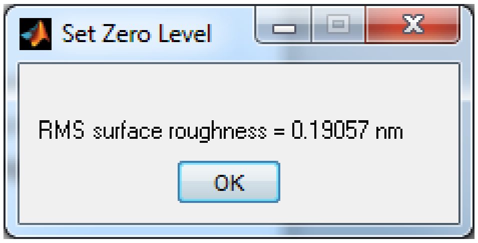


Figure S6. Correction of uneven image illumination. (A) Typical image profile shown as a red line after polynomial flattening in NanoScope software. (B) Result of the **Remove Surface** processing (top-hat filter) based on erosion and dilation morphological operations, but with the violated average height of the surface in absolute values. (C) By using Set Zero Level procedure one can define a background area (dashed gray line) in the image and assign zero height level to the mean intensity inside that region.

With the last action the program also automatically calculates the root mean square (RMS) surface roughness $R_{\mathrm{RMS}} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} h_i^2}$ and provides this value in a dialog box (Figure S7) after completing the leveling. It may serve as an uncertainty in height estimation upon tracking, since fibrils lay on top of a "bumpy carpet".



RMS surface roughness = 0.19057 nm

Figure S7. Root mean square (RMS) surface roughness dialog box appears after setting the zero level.

Fiber tracking

Tracking manually a decent amount of fiber-like objects is quite often a "bottleneck" in a whole statistical analysis procedure for any given system. Many automatic segmentation methods exist to address that issue; however, all of them have certain limitations such that some systems cannot be processed by them. To get meaningful results these methods require not only a high quality of imaging, but also the presence of only one particular type of objects (i.e. with similar height or flexibility) and/or absence of considerably overlapping fibrils. Here we tried to keep a very general approach, suitable for a full range of images and fiber-like object types. The semi-automatic tracking seems to be an excellent compromise, which keeps a low complexity to efficiency ratio of programming implementation. As a rule for crowded or low resolution images we define the following concept: "if one can follow the whole fiber along its contour with bare eyes, then the tracking algorithm should be able to obtain coordinates of its middle line". To pursue it we combine two complementary tracking algorithms that bypass weaknesses of each another: A* pathfinding algorithm[2] for a rough preliminary tracking and Active Contour Models energy minimization algorithm for the sub-pixel accuracy[3,4].

Active Contour Models (Active Contours)

The Active Contour Model can be described as a discrete contour with the internal energy potential penalizing bending between consequent segments, while being placed into an external force field, defined by the image gradient, is guided towards the bright central line of a fiber-like object. The internal energy potential causes the Active Contour to be straight but still flexible, smoothing the abrupt noise influence from the image. The internal energy term, E_{int}, is defined similar to closed snakes for discrete contours described by $\vec{r}_i = (x_i, y_i)$:

$$E_{\text{int}} = \sum_i \alpha_i \left|\vec{t}_i\right|^2 + \beta_i |\vec{\kappa}_i|^2,$$

where κ_i is the curvature at the point i, $\vec{\kappa}_i = \frac{\vec{t}_{i+1}-\vec{t}_i}{\Delta s}$, $\vec{t}_i$ is the tangent vector to the contour $\vec{t}_i = \frac{\vec{r}_{i+1}-\vec{r}_i}{\Delta s}$, and α_i and β_i are coefficients that control intrinsic mechanical properties. For the Open Active Contours it is reasonable to keep $\alpha_i = 0 \; \forall i$, and it can be justified by a form of the internal energy of Worm-like Chain (WLC) model:[5]

$$E_{\text{WLC}} = \frac{b}{2} \sum_i \Delta s \, \kappa_i^2$$

where b is the bending rigidity.

The external energy, E_{ext}, consists of two terms: an image term, E_{img}, and a stretching term, E_{str} are defined as:[4]

$$E_{\text{ext}} = \sum_{i=1}^{N} \left(\kappa_1 E_{\text{img}} + \kappa_2 E_{\text{str}}\right)$$

$$E_{\text{img}}\left(\vec{r}(s)\right) = \frac{\vec{\nabla} I}{I_f}$$

$$E_{\text{str}}^{1,N} = \frac{I^{1,N}}{I_f}$$

The active-contour-based segmentation works by minimizing the contour's overall energy, E, which is composed of the internal energy, E_{int}, and the external energy, E_{ext}, i.e.,

$$E = E_{\text{int}} + E_{\text{ext}}$$

Using the Euler's method to minimize the overall energy function E:

$$\begin{cases} x_t = (A + \gamma I)^{-1}\left(\gamma x_{t-1} - f_x(x_{t-1}, y_{t-1})\right) \\ y_t = (A + \gamma I)^{-1}\left(\gamma y_{t-1} - f_y(x_{t-1}, y_{t-1})\right) \end{cases}$$

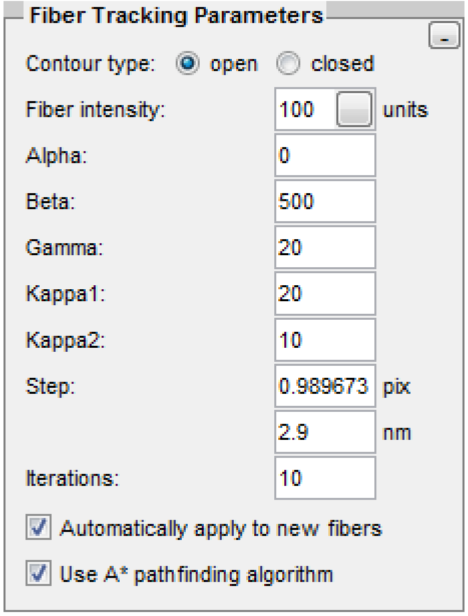
where $f_x = \partial E_{\text{ext}} / \partial x$ and $f_y = \partial E_{\text{ext}} / \partial y$.

Using finite difference formulas, the pentadiagonal matrix A has the form:

$$\begin{cases} A(i, i-2) = \beta_{i-1} \\ A(i, i-1) = -\alpha_i - 2[\beta_{i-1} + \beta_i] \\ A(i, i) = \alpha_i + \alpha_{i+1} + \beta_{i-1} + 4\beta_i + \beta_{i+1} \\ A(i, i+1) = -\alpha_{i+1} - 2[\beta_i + \beta_{i+1}] \\ A(i, i+2) = \beta_{i+1} \end{cases}$$

Setting Fiber Tracking Parameters

This panel can be accessed *via* **FiberApp** → **Main Panels** → **Fiber Tracking Parameters** item, or *via* the toolbar toggle button .



Fiber Tracking Parameters

Contour type: ● open ○ closed

Fiber intensity: 100 units

Alpha: 0

Beta: 500

Gamma: 20

Kappa1: 20

Kappa2: 10

Step: 0.989673 pix

2.9 nm

Iterations: 10

☑ Automatically apply to new fibers

☑ Use A* pathfinding algorithm

Figure S8. Fiber Tracking Parameters panel.
Contour type: open – both ends are free, closed – both ends are connected to each other.
Fiber intensity – value in image intensity units of desired type of fibers.
Alpha – parameter α.
Beta – parameter β.
Gamma – parameter γ.
Kappa1 – parameter κ_1.
Kappa2 – parameter κ_2.
Step (Δs) – step size in pixels (top field) and nanometers (bottom field).
Iterations – number of times fitting algorithm is applied while pressing **Fit Fiber**.
Automatically apply to new fibers – fitting algorithm will be applied as many times as defined in Iterations field to all newly tracked fibers.
Use A* pathfinding algorithm – initiate tracking with the rough estimation of a fibers midline using A* pathfinding algorithm.

A* pathfinding algorithm

Active Contour Models are not capable of solving the entire problem of finding fiber-like objects in an image. These contours need to be initialized close enough to the desired energy local minima. Because in general no satisfactory initialization exists due to the complex task-dependent nature of image processing, these contours can still be used for the semi-automatic image interpretation. The algorithm only provides assistance for a user who defines an image feature to be tracked (Figure S9). The implemented algorithm has one difference from the classical A*: the cost-growth upon moving through a pixel is inversely proportional to that pixel intensity.

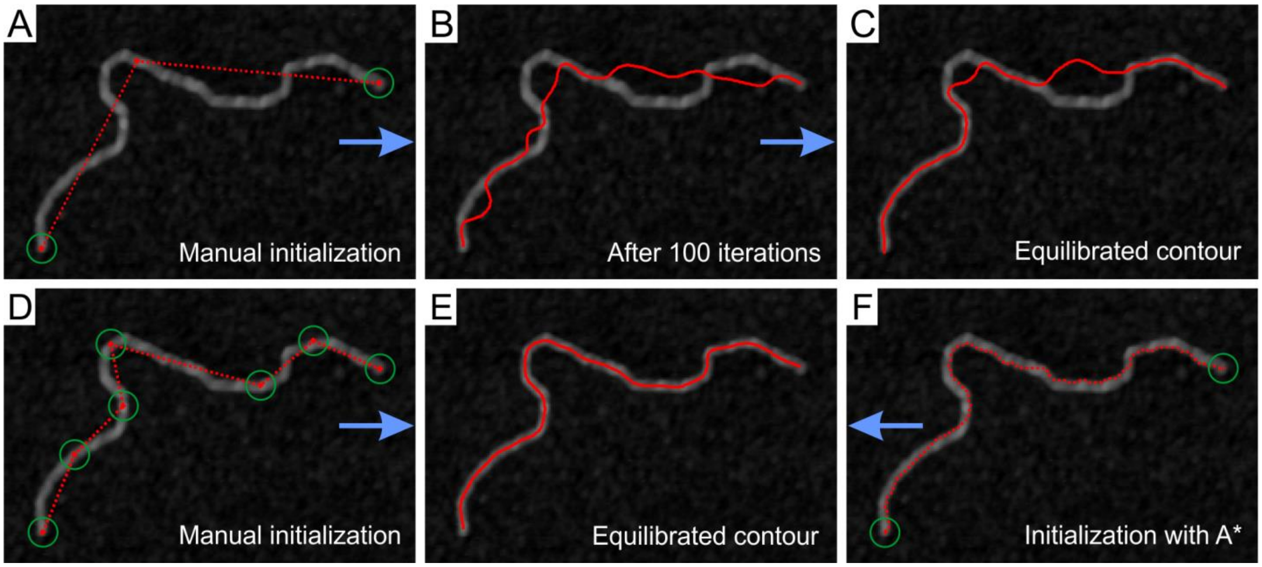
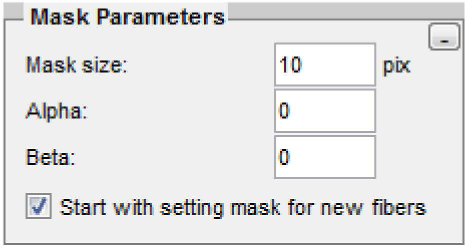


Figure S9. Fiber tracking procedure using A* pathfinding and Active Contours algorithms. (A) Rough manual placing of the points without assistance of A*. (B) The contour after 100 iterations of Active Contour Models energy minimization. (C) Final and equilibrated position of the contour, algorithm failed to find a desired solution due to rough initialization of the contour. (D) Accurate manual initialization of the contour requires 1.5-3 times more intermediate user clicks, which is very unwanted. However, the desirable result is reached within 100 iterations of the fitting algorithm (E). (F) The same result can be reached with help of A* pathfinding algorithm, which allows an initialization of the contour with only 2 user clicks on both ends of the fiber-like object.

Some of the useful shortcuts for using the program:
(Space), toolbar button ➕, or **Fiber Tracking** → **Add Fiber** menu item – to add a new fiber.
(F), toolbar button ➡, or **Fiber Tracking** → **Fit Fiber** menu item – to apply the fitting algorithm.
(Delete), toolbar button ➖, or **Fiber Tracking** → **Remove Fiber** menu item – to delete the currently selected fiber.
(Backspace) upon tracking – delete the last point of a contour.
Arrow keys – pan the image.

Tracking objects with nonhomogeneous bending properties using masks



Mask Parameters

Mask size: 10 pix

Alpha: 0

Beta: 0

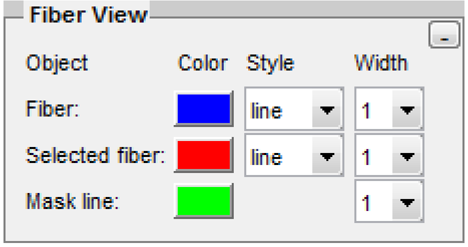
☑ Start with setting mask for new fibers

Figure S10. Mask Parameters panel.
Mask size – box size in pixels of a square mask element.
Alpha – parameter α inside mask areas.
Beta – parameter β inside mask areas.
Start with setting mask for new fibers – initiate by placing mask elements prior to fiber contour tracking.

This panel can be accessed *via* **FiberApp** → **Main Panels** → **Mask Parameters** item, or *via* the toolbar toggle button ▨.

Appearance of the masks and tracked objects

The color, style and line width of tracked fibers and mask elements can be modified for better visual appearance using **Fiber View** panel. The possible style variants for fibers are “points” and “line”. In the first case fibers are represented by point-like markers at xy positions of fiber nodes and, in the second case, by a continuous polygonal line that is composed of straight segments connecting these nodes.



Fiber View

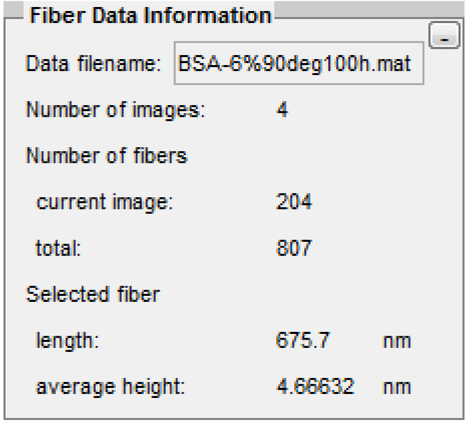
Object	Color	Style	Width
Fiber:	🟦	line	1
Selected fiber:	🟥	line	1
Mask line:	🟩		1

Figure S11. Fiber View panel.
Fiber – visual properties of the unselected tracked fibers.
Selected fiber – visual properties of currently selected tracked fiber.
Mask line – visual properties of the mask elements.

This panel can be accessed *via* **FiberApp** → **Main Panels** → **Fiber View** item, or *via* the toolbar toggle button 👁.

Fiber Data file

The fibers data can be saved on the computer disk as a project MAT file by pressing either **FiberApp** → **Save Data**, the toolbar button 💾, or **FiberApp** → **Save Data As** for saving under a different file name. The stored data of tracked fibers can be loaded into the software by choosing **FiberApp** → **Open Data**, or *via* the toolbar item 📂.



Fiber Data Information

Data filename: BSA-6%90deg100h.mat

Number of images: 4

Number of fibers

current image: 204

total: 807

Selected fiber

length: 675.7 nm

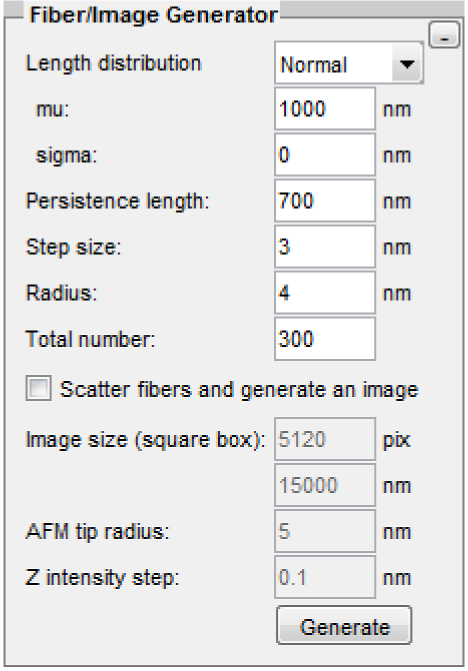
average height: 4.66632 nm

Figure S12. Fiber Data Information panel.
Data filename – name of the file on the disk where the current fiber data is stored.
Number of images indicates the number of processed images from which the tracked contours are stored in the data file.
Number of fibers tracked in the current image and in total and stored in the current data file.
Selected fiber basic characteristics: length and average height in nanometers.

This panel can be accessed *via* **FiberApp** → **Main Panels** → **Fiber Data Information** item, or *via* the toolbar toggle button 📋.

Fiber/Image Generator

An angle θ between the two successive segments with length Δs of simulated fibrils is chosen from a Gaussian distribution centered at 0 and of variance σ_θ: $\sigma_\theta = \sqrt{\Delta s/\lambda}$.



Length distribution: Normal

mu: 1000 nm

sigma: 0 nm

Persistence length: 700 nm

Step size: 3 nm

Radius: 4 nm

Total number: 300

□ Scatter fibers and generate an image

Image size (square box): 5120 pix

15000 nm

AFM tip radius: 5 nm

Z intensity step: 0.1 nm

Figure S13. Fiber/Image Generator panel.
Length distribution: Normal (Gaussian) or Lognormal, where
mu (μ) – mean value;
sigma (σ) – standard deviation.
Persistence length of generated contours.
Step size (Δs) – distance between two consecutive points along generated contours.
Radius – radius of generated fibers.
Total number – total number of fibers.
Scatter fibers and generate an image – when this checkbox is selected, the program will also generate a TIF image with randomly distributed fibers. The next parameters related to the image generation will be available to be modified.
Image size (square box) – image size in pixels (top field) and nanometers (bottom field).
AFM tip radius – simulated WLCs will be widened according to the approximation of morphological dilation by the AFM tip.
Z intensity step – the minimal change in intensity and corresponding change in height.

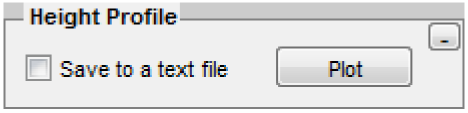
This panel can be accessed *via* **Tools** → **Fiber/Image Generator** item, or *via* the toolbar toggle button .

Data Analysis

All methods for data analysis are represented as panels in *FiberApp*, and can be found under **Processing** menu item. Each panel requires a fiber data to be loaded in the software on which this particular analysis will be carried out. It is also not necessary to open the corresponding image, only the fiber data is required (the only exception is **Height Profile** panel). Panels produce a graphical output (plots, distributions, profiles), which can be saved as an ASCII text file and used in a 3rd party programs like Origin for plotting with a control over visual appearance. To do so, tick Save to a text file checkbox before implementing the panel action. After completing panels' function, the user will be prompted to save results in a file with the desired name and location on the computer. Examples of the output from the following methods can be found in the main text.

Height Profile

This panel can be used to plot height profiles along curved objects, as well as linear cross-sections. For AFM images the *z*-scale parameter has a certain value, thus, the fiber's height will be in nanometers, and in intensity units for other types of images. Discretization is equal to the tracking step Δs.



Height Profile

☐ Save to a text file

Figure S14. Height profile panel.

This panel can be accessed *via* **Processing** → **Height Profile** menu item.

Because the points of the tracked contour do not necessarily lie in the pixel centers, all height values are interpolated based on surrounding pixels. The possible interpolation methods and corresponding height profiles are shown in Figure S14. The default interpolation method is bicubic and can be changed in **Fiber Tracking** → **Z interpolation** menu item.

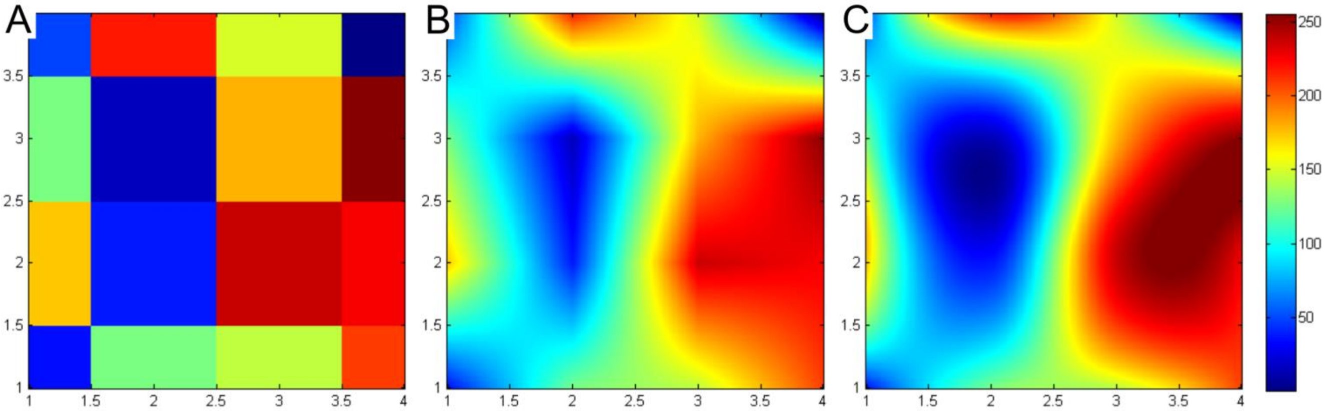
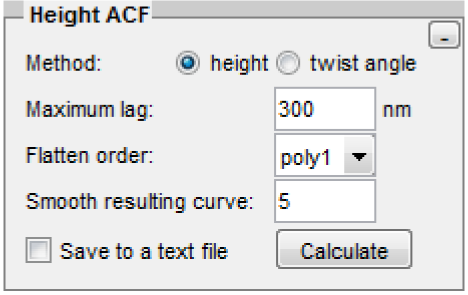


Figure S14. Interpolation methods in 2D. (A) Nearest neighbor, (B) bilinear interpolation, and (C) bicubic interpolation (default in *FiberApp*).

Height Autocorrelation Function (Height ACF)



Height ACF

Method: ● height ○ twist angle

Maximum lag: 300 nm

Flatten order: poly1

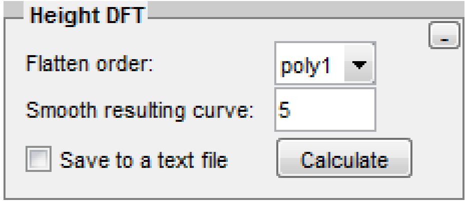
Smooth resulting curve: 5

☐ Save to a text file

Figure S15. Height ACF panel.
Method: height or twist angle.
Maximum lag – the largest relative shift of the high profile in the autocorrelation function
Flatten order: none – no subtraction, const – subtract a constant value C, poly1 – subtract a function $Ax + C$, poly2 – subtract a function $Ax^2 + Bx + C$.
Smooth resulting curve using “moving average” method.

This panel can be accessed *via* **Processing** → **Height ACF** menu item.

Height Discrete Fourier Transform (Height DFT)



Height DFT

Flatten order: poly1

Smooth resulting curve: 5

☐ Save to a text file

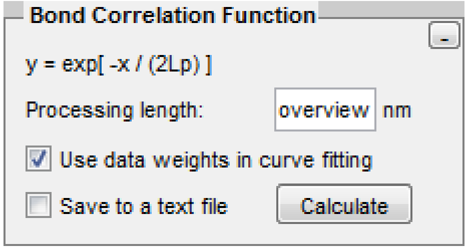
Figure S16. Height DFT panel.
Flatten order: none – no subtraction, const – subtract a constant value C, poly1 – subtract a function $Ax + C$, poly2 – subtract a function $Ax^2 + Bx + C$.
Smooth resulting curve using “moving average” method.

This panel can be accessed *via* **Processing** → **Height DFT** menu item.

Methods of a persistence length estimation

The persistence length is a basic mechanical property of polymers, which quantifies its rigidity and is formally defined *via* the bond correlation function as the length over which the angular correlations in the tangent direction decrease by *e* times.

Bond Correlation Function


Bond Correlation Function

y = exp[-x / (2Lp)]

Processing length: overview nm

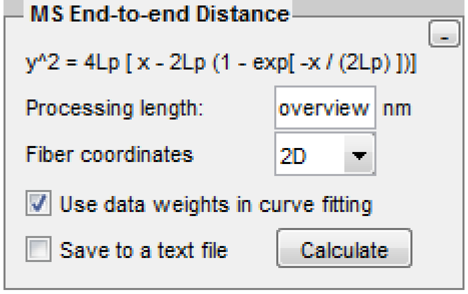
☑ Use data weights in curve fitting

□ Save to a text file

Figure S17. Bond Correlation Function panel.
Processing length – maximal separation (*s*) value.
Use data weights in curve fitting, where weights are proportional to the number of calculation counts at a particular separation.

This panel can be accessed *via* **Processing** → **Bond Correlation Function** menu item.

Mean-Squared End-to-end Distance


MS End-to-end Distance

y^2 = 4Lp [x - 2Lp (1 - exp[-x / (2Lp)])]

Processing length: overview nm

Fiber coordinates 2D

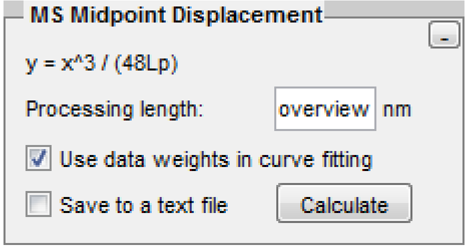
☑ Use data weights in curve fitting

□ Save to a text file

Figure S18. MS End-to-end Distance panel.
Processing length – maximal separation (*s*) value.
Fiber coordinates: 2D – fibers are equilibrated on a surface, Projection – tracked fibers are projections on a surface of real 3D structures.
Use data weights in curve fitting, where weights are proportional to the number of calculation counts at a particular separation.

This panel can be accessed *via* **Processing** → **MS End-to-end Distance** menu item.

Mean-Squared Midpoint Displacement


MS Midpoint Displacement

y = x^3 / (48Lp)

Processing length: overview nm

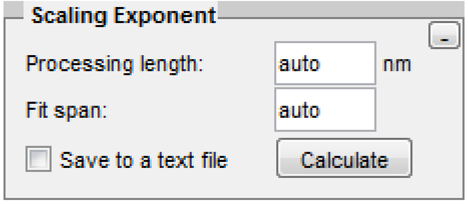
☑ Use data weights in curve fitting

□ Save to a text file

Figure S19. MS Midpoint Displacement panel.
Processing length – maximal separation (*s*) value.
Use data weights in curve fitting, where weights are proportional to the number of calculation counts at a particular separation.

This panel can be accessed *via* **Processing** → **MS Midpoint Displacement** menu item.

Scaling Exponent


Scaling Exponent

Processing length: auto nm

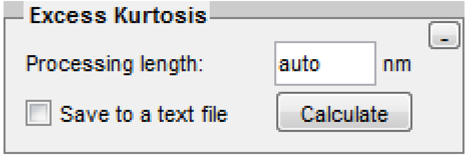
Fit span: auto

□ Save to a text file

Figure S20. Scaling Exponent panel.
Processing length – maximal separation (*s*) value.
Fit span – a window of fitting to a linear function in order to extract the scaling exponent.

This panel can be accessed *via* **Processing** → **Scaling Exponent** menu item.

Excess Kurtosis


Excess Kurtosis

Processing length: auto nm

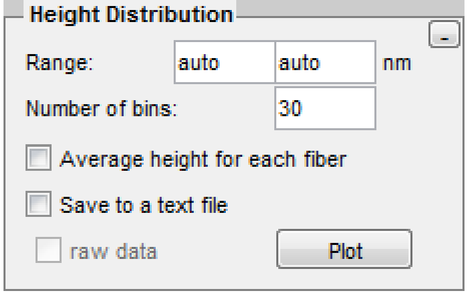
□ Save to a text file

Figure S21. Excess Kurtosis panel.
Processing length – maximal separation (*s*) value.

This panel can be accessed *via* **Processing** → **Excess Kurtosis** menu item.

Height Distribution

Two different and complementary to each other height distributions are computed here. The first is a total height distribution of all points among all fibers. The second is a distribution of averaged fibers heights.


Height Distribution

Range: auto auto nm

Number of bins: 30

□ Average height for each fiber

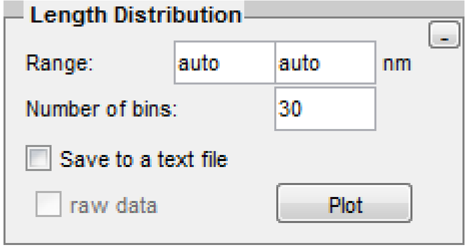
□ Save to a text file

□ raw data

Figure S22. Height Distribution panel.
Range – minimal and maximal values of the distribution.
Number of bins – discretization of the distribution.
Average height for each fiber – plot the averaged fiber heights histogram instead of height distribution of all points among all fibers.
Raw data – save height values instead of histogram bin counts and their positions.

This panel can be accessed *via* **Processing** → **Height Distribution** menu item.

Length Distribution


Length Distribution

Range: auto auto nm

Number of bins: 30

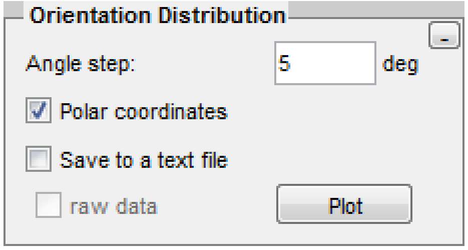
□ Save to a text file

□ raw data

Figure S23. Length Distribution panel.
Range – minimal and maximal values of the distribution.
Number of bins – discretization of the distribution.
Raw data – save length values instead of histogram bin counts and their positions.

This panel can be accessed *via* **Processing** → **Length Distribution** menu item.

Orientation Distribution


Orientation Distribution

Angle step: 5 deg

☑ Polar coordinates

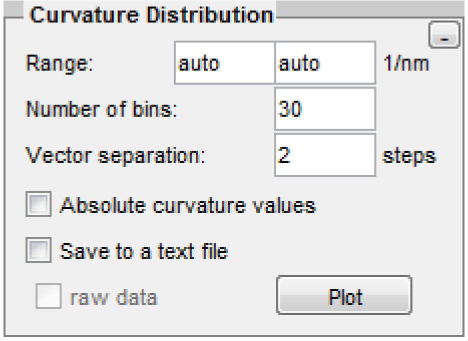
□ Save to a text file

□ raw data

Figure S24. Orientation Distribution panel.
Angle step – discretization of the distribution.
Polar coordinates – plot the distribution in polar coordinates.
Raw data – save orientation angle values instead of histogram bin counts and their positions.

This panel can be accessed *via* **Processing** → **Orientation Distribution** menu item.

Curvature Distribution



Curvature Distribution

Range: auto auto 1/nm

Number of bins: 30

Vector separation: 2 steps

□ Absolute curvature values

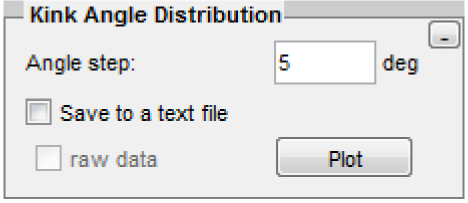
□ Save to a text file

□ raw data

Figure S25. Curvature Distribution panel.
<u>Range</u> – minimal and maximal values of the distribution.
<u>Number of bins</u> – discretization of the distribution.
<u>Vector separation</u> – the curvature will be calculated based on segments, separated by the number of steps.
<u>Absolute curvature values</u> – plot the absolute curvature values.
<u>Raw data</u> – save curvature values instead of histogram bin counts and their positions.

This panel can be accessed *via* **Processing** → **Curvature Distribution** menu item.

Kink Angle Distribution



Kink Angle Distribution

Angle step: 5 deg

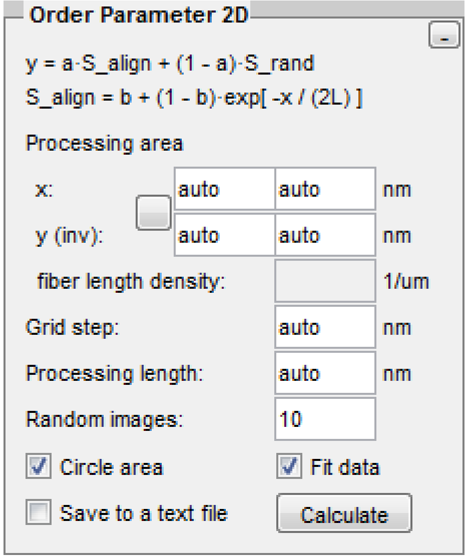
□ Save to a text file

□ raw data

Figure S26. Kink Angle Distribution panel.
<u>Angle step</u> – discretization of the distribution.
<u>Raw data</u> – save curvature values instead of histogram bin counts and their positions.

This panel can be accessed *via* **Processing** → **Kink Angle Distribution** menu item.

Order parameter 2D



Order Parameter 2D

y = a·S_align + (1 - a)·S_rand

S_align = b + (1 - b)·exp[-x / (2L)]

Processing area

x: auto auto nm

y (inv): auto auto nm

fiber length density: 1/um

Grid step: auto nm

Processing length: auto nm

Random images: 10

☑ Circle area ☑ Fit data

□ Save to a text file

Figure S27. Order Parameter 2D panel.
Processing area:

<u>x</u> – area range along horizontal axis.
<u>y (inv)</u> – area range along vertical axis.
<u>fiber length density</u> inside the defined area.

<u>Grid step</u> – minimal box size and also discretization of the final plot.
<u>Processing length</u> – maximal box size.
<u>Random images</u> – number of generated images for the S_{2D}^{rand} evaluation.
<u>Circle area</u> – use circle area instead of square.
<u>Fit data</u> – provide the fit and estimate *a* and *b* parameters.

This panel can be accessed *via* **Processing** → **Order Parameter 2D** menu item.

2D Order Parameter (S_{2D}) definition is:

$$S_{2D} = 2\langle\cos^2\theta_n\rangle - 1 = \langle\cos(2\theta_n)\rangle$$

Let us consider our fibril segments as vectors with normalized coordinates with respect to an arbitrary Cartesian coordinate system: $\{x_1; y_1\}, \dots, \{x_N; y_N\}$, ${x_n}^2 + {y_n}^2 = 1$, $n = 1, \dots, N$. An unknown director is a vector and has normalized coordinates: $\{x_d; y_d\}$, ${x_d}^2 + {y_d}^2 = 1$. According to the scalar product formula: $\cos\theta_n = x_n x_d + y_n y_d$, and due to the fact, that within all possible directions of the director we should choose one which will maximize the value of the 2D Order Parameter, we can rewrite the main equation:

$$S_{2D} = 2\left[\frac{1}{N}\sum_{n=1}^{N}(x_n x_d + y_n y_d)^2\right] - 1 \rightarrow \max$$

Expanding the brackets:

$$S_{2D} = 2\left[x_d^2 \cdot \frac{1}{N}\sum_{n=1}^{N} x_n^2 + 2x_d y_d \cdot \frac{1}{N}\sum_{n=1}^{N} x_n y_n + y_d^2 \cdot \frac{1}{N}\sum_{n=1}^{N} y_n^2\right] - 1 \rightarrow \max$$

Making the following changes of variables:

$$\frac{1}{N}\sum_{n=1}^{N} x_n^2 = A, \quad \frac{1}{N}\sum_{n=1}^{N} x_n y_n = B,$$

$$x_d = \cos\alpha\,, y_d = \sin\alpha$$

where α is a unknown angle at which the director is tilted to our coordinate system.

The main equation takes the form:

$$S_{2D} = (2A - 1)\cos 2\alpha + 2B\sin 2\alpha \rightarrow \max$$

From trigonometric formula: $C_1 \sin\gamma + C_2\cos\gamma = \sqrt{C_1^2 + C_2^2}\,\sin(\gamma + \vartheta)$, where $\cos\vartheta = \frac{C_1}{\sqrt{C_1^2 + C_2^2}}$ we can finally come to a form with one trigonometric function.

$$S_{2D} = \sqrt{(2A-1)^2 + (2B)^2}\,\sin(2\alpha + \vartheta) \rightarrow \max$$

The period of this formula is a function of α is π, because a 2D order parameter is independent of the direction of the director, which has two equal values for opposite directions. With the help of the fact that we should choose the director in a way to maximize function value we can get rid of both the unknown angles.

Hence, $S_{2D} = \sqrt{(2A-1)^2 + (2B)^2}$, where

$$A = \frac{1}{N}\sum_{n=1}^{N} x_n^2,$$

$$B = \frac{1}{N}\sum_{n=1}^{N} x_n y_n.$$

References

(1) Dougherty, E. R. *An introduction to morphological image processing*; SPIE Optical Engineering Press, 1992.

(2) Hart, P.; Nilsson, N.; Raphael, B. *IEEE Trans. Syst. Sci. Cybern.* **1968**, *4*, 100–107.

(3) Kass, M.; Witkin, A.; Terzopoulos, D. *Int. J. Comput. Vis.* **1988**, *1*, 321–331.

(4) Smith, M. B.; Li, H.; Shen, T.; Huang, X.; Yusuf, E.; Vavylonis, D. *Cytoskeleton (Hoboken).* **2010**, *67*, 693–705.

(5) Rappaport, S. M.; Medalion, S.; Rabin, Y. *arXiv:0801.3183* **2008**.