Supplemental Material: detectCilia: An R Package for Automated Detection and Length Measurement of Primary Cilia

Workflow for manually finding and measuring cilia

The following steps should be performed:

1. Open the original confocal microscopy image (CZI files) in Fiji/ImageJ using the settings shown in Figure S1.
2. Close channel 0 (nuclei in blue) and channel 1 (actin in green). Channel 2 contains the cilia pixels in red.
3. Detect cilia:
   - 3.1 Create a z-stack projection: `Image` → `Stacks` → `Z Project...` → `Max Intensity`.
   - 3.2 Adjust brightness and contrast of the z-stack projection: `Image` → `Adjust` → `Brightness and Contrast` → `Auto` → `Apply`.
   - 3.3 Number all structures identified to be cilia ignoring those touching a border (x or y) (see Figure S2A): `Text tool` → enter a number → [Ctrl] + [B].
4. Measure horizontal cilia lengths (in z-stack projection):
   - 4.1 Delete scale to only measure lengths in pixels and not in µm: `Analyze` → `Set scale...` → `Click to remove scale` → `OK`.
   - 4.2 Use a straight line and connect two points of every cilium such that the longest distance is being obtained (see Figure S2B): `Straight line` → Click and hold to connect two points → [Ctrl] + [M] to measure the distance → Save all results as CSV.
5. Measure vertical cilia lengths (in z stack)
   - 5.1 Go back to the z-stack image.
   - 5.2 Adjust brightness and contrast of the z stack: `Image` → `Adjust` → `Brightness and Contrast` → `Auto` → `Apply`.
   - 5.3 Zoom to every cilium (`Magnifying glass`) and look for the lowest and highest z-stack layers that show pixels that can be identified to belong to the cilium (see Figure S2C and D). Record these layers.
6. The total lengths of the cilia are calculated with an R script applying Pythagoras' theorem.

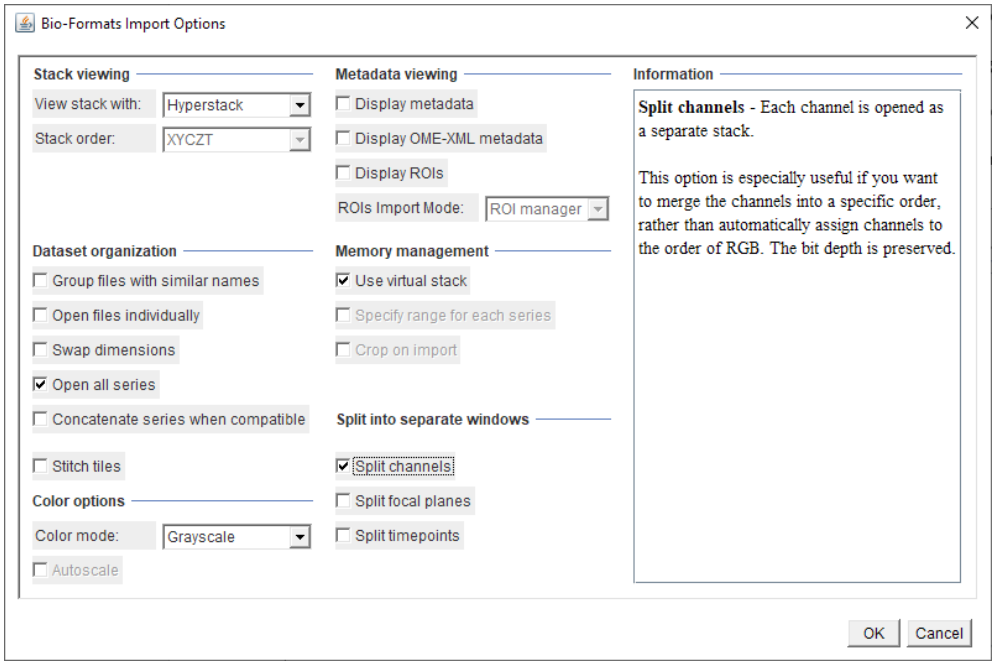

Figure S1. Settings in Fiji to import an image in the CZI file format.

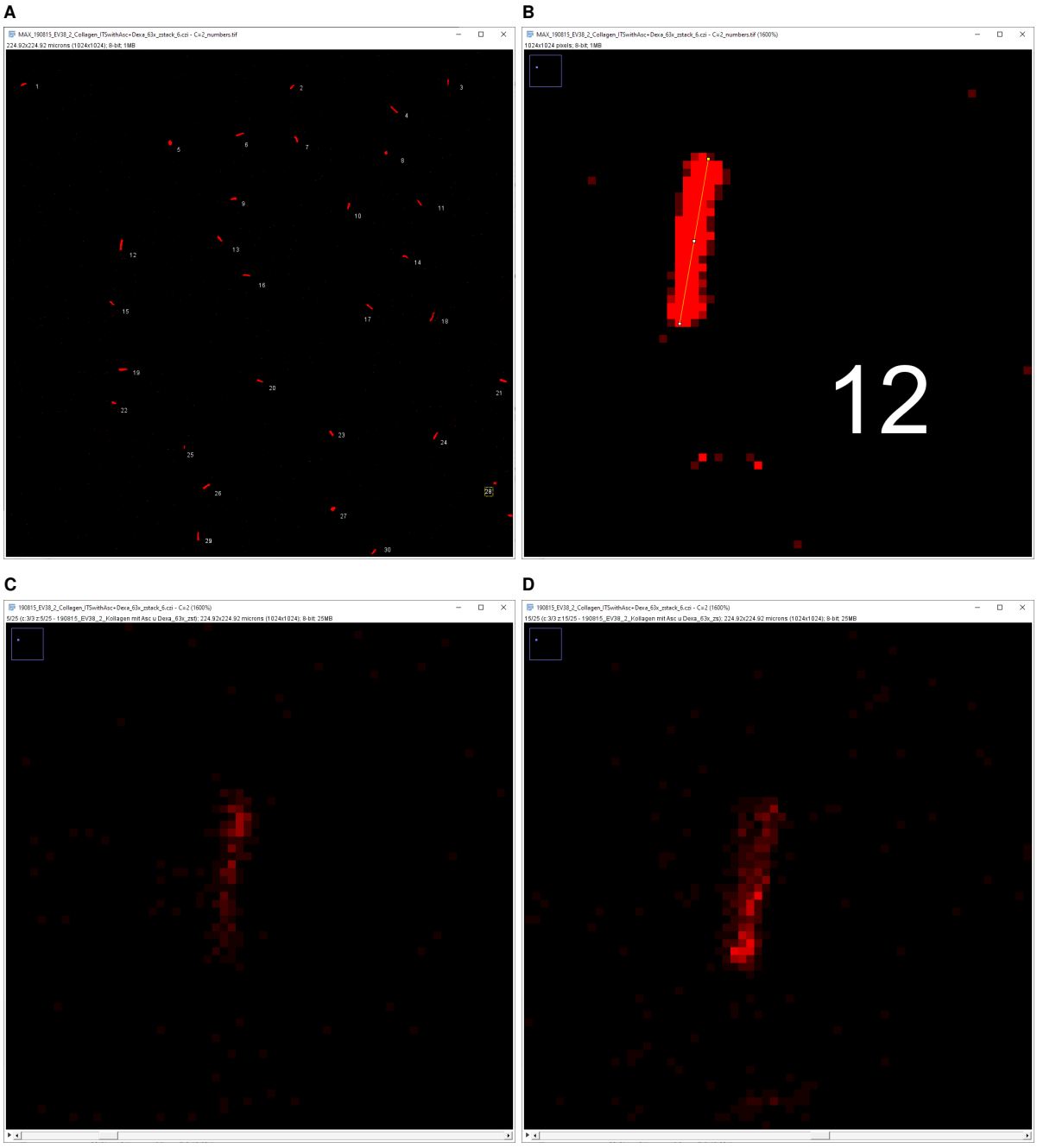

Figure S2. Example for manually detecting and measuring cilia in `190815_EV38_2_Collagen_ITSwithAsc+Dexa_63x_zstack_1.czi` using Fiji/ImageJ. **A**: All structures identified as a cilium are numbered in the z-stack projection. **B**: The horizontal length is measured with the line tool in the z-stack projection. **C** and **D**: The lower (*z=5*) and upper limit (*z=15*), and thus the vertical length, is determined by going through the z stacks and finding the first and last layer where parts of a cilium are seen.

Difficulties in identifying primary cilia

Manual and automated detection of cilia can never be done without any uncertainties. Two example images are shown in Figure S3.

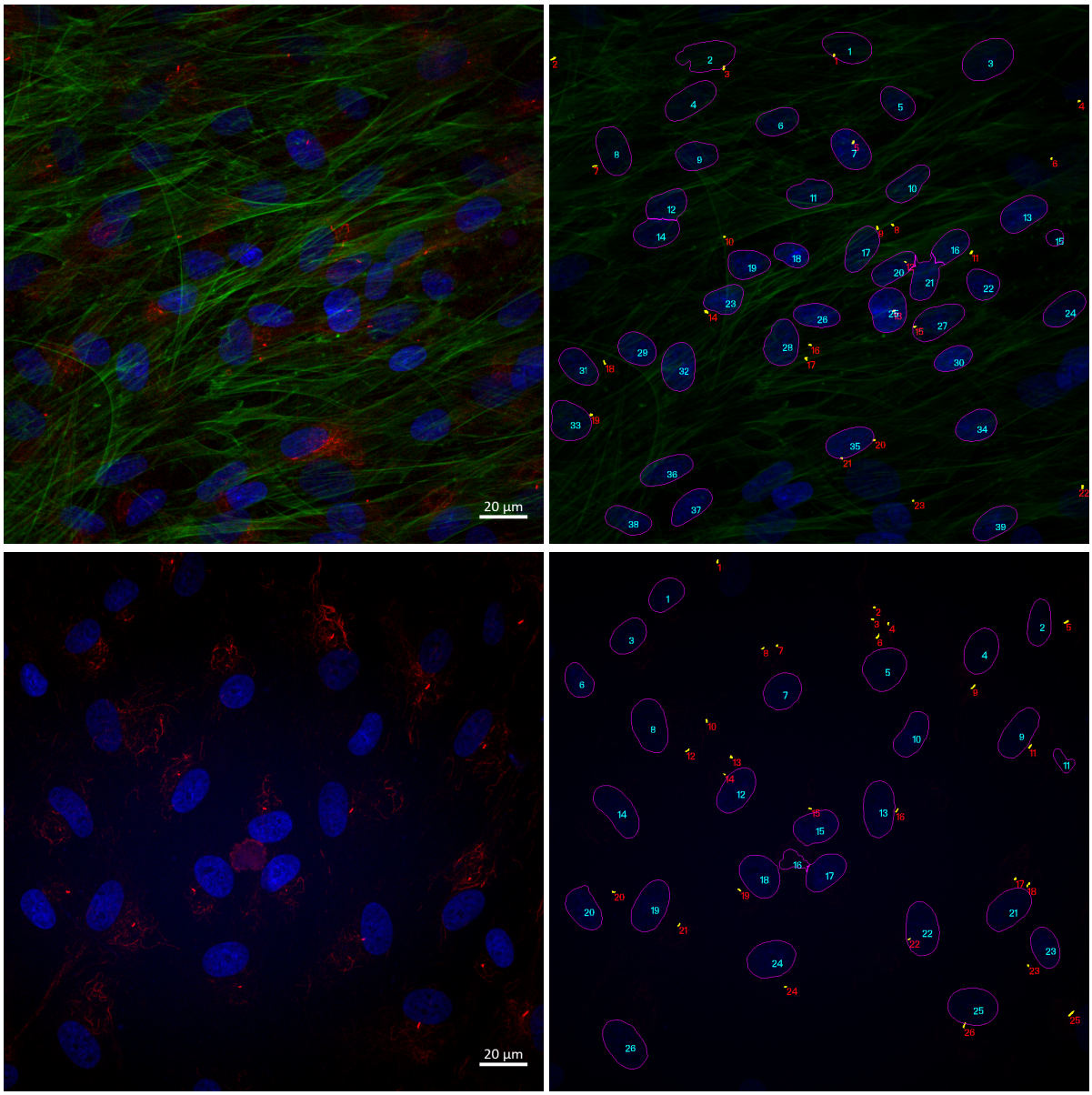

Figure S3. Example images showing lots of "noise" in the red channel containing cilia (left) as well as the results from detectCilia (right). The brightness and contrast of the images have been manually modified for better visibility. First row: z-stack projection of the image `190808_EV38_1_Collagen_FBSwithAsc_63x_zstack_3.czi`. Second row: z-stack projection of the image `210301_EV38_1_Collagen_ITSwithAsc+Dexa+IGF+TGF_63x_z-stack_3_1024x1024.czi`. (There is no green layer in this image.)

R package detectCilia

In the following, we briefly explain the input parameters of the main function *detectCilia()*.

Input parameters of the main function detectCilia()

The main function for cilia detection needs the following input:

- `input_dir_tif` or `input_file_czi`: Either a directory with `TIF` files of one (confocal) microscopy image (one `TIF` file for every z-stack layer) or one `CZI` file to be read.
- `output_dir`: Path to a directory where the results are to be saved.
- `cilium_color`: Color of the channel used to detect the secondary antibody attached to the primary antibody of the primary cilium. The default value is "red".
- `nucleus_color`: Color that corresponds to the channel depicting the stained nuclei (using, for example, Hoechst or DAPI). The default value is "blue".
- `projection_method_for_threshold_calculation`: This value determines how the z-stack projection is realized, which is needed for threshold calculations. The methods "mean" or "max" may be used. The default value is "max".
- `use_histogram_equalization_for_threshold_calculation`: This states whether to use a histogram equalization algorithm for determining threshold values and finding cilia in projections. The default value is "FALSE".
- `threshold_by_density_of_cilium_pixels`: This parameter states whether the thresholds that depend on the histogram of the cilium color channel or custom thresholds (see the following two parameter values) are to be used. The default value is "TRUE", which disregards possible custom threshold values.
- `threshold_find`: It corresponds to the minimum intensity (a number between 0 and 1) of possible cilia pixels in the z-stack projection. The default value for considering a pixel from the z-stack projection to be part of a cilium is "0.01" when `threshold_by_density_of_cilium_pixels` is given to be "FALSE" and no threshold value is explicitly provided. If `threshold_by_density_of_cilium_pixels` is "TRUE", then

$$\texttt{threshold\_find} = quantile(\text{Image\_projection}, (1 - \text{ratio\_of\_cilia\_pixels})) \tag{1}$$

  with

$$\text{ratio\_of\_cilia\_pixels} = R_{cp} = \text{number\_of\_nuclei} \cdot \text{average\_cilium\_area}/\text{area\_of\_image\_stack} \tag{2}$$

  and `Image_projection` being the z-stack projection. (The R function "quantile" returns estimates of underlying distribution quantiles.)
- `threshold_connect`: The default value for considering a pixel from every z-stack layer to be part of a cilium is "0.005" when `threshold_by_density_of_cilium_pixels` is given to be "FALSE" without specifying the threshold. If `threshold_by_density_of_cilium_pixels` is "TRUE", `threshold_connect` is $0.1 \cdot$ `threshold_find`.
- `vicinity_combine`: It determines how far away another detected cilium pixel may be to be assigned to the same cilium of the z-stack projection. If no explicit value is specified, it will be determined based on the result of ceiling$(0.5 \cdot$ sqrt$($`max_cilium_area_in_pixels`$))$. The default value is "NULL".
- `vicinity_connect`: If no explicit value is specified, it will be determined based on the result of floor$(0.5 \cdot$ sqrt$($`min_cilium_area_in_pixels`$))$. It determines how far away another detected cilium pixel may be to be assigned to a found cilium in every z-stack layer. The default value is "NULL".
- `max_cilium_area_in_pixels`: If no explicit value is specified, the exact value will be determined based on the `pixel_size` and the assumption that a cilium is smaller than $5\,\mu\text{m} \times 1\,\mu\text{m}$. It determines the maximum number of pixels allowed for a detected structure to be interpreted as a cilium. The default value is "NULL".
- `min_cilium_area_in_pixels`: If no explicit value is specified, the exact value will be determined based on the `pixel_size` and the assumption that a cilium is larger than $1\,\mu\text{m} \times 0.25\,\mu\text{m}$. It determines the minimum number of pixels required for a detected structure to be interpreted as a cilium. The default value is "NULL".
- `number_size_factor`: This parameter determines the size of the printed numbers on the exported images. If no explicit value is specified, it will be determined depending on the dimension of the image. The default value is "NULL".
- `pixel_size`: When CZI images are used, the pixel size is read from the metadata. When TIF images are used, this value needs to be given (in μm) to determine the horizontal cilia length. The default value is "NULL".

- `slice_distance`: When CZI images are used, the distance of the z-stack layers is read from the metadata. When TIF images are used, this value must be given (in µm) to determine the vertical cilia length. The default value is "NULL".
- `nuc_mask_width_height_in_pixels`: It determines the width and height of the moving adaptive thresholding rectangle for detecting nuclei. If no explicit value is specified, the exact value will be determined based on the `pixel_size` and the assumed nuclei area of $15\,\mu m \times 15\,\mu m$. The default value is "NULL".
- `export_normalized_images`: If "TRUE", normalized versions of the z-stack projection and of the z-stack layers are exported. The default value is "FALSE".
- `number_of_expected_nuclei`: This parameter is needed for calculating thresholds if no nuclei are present in the image, as we expect one cilium (max.) per cell. The default value is "NULL".

Return values of the main function detectCilia()

The function output is

- a list containing different data frames: `df_parameterlist`, `df_cilium_information`, `df_cilium_summary` (see Section Main steps of the main function *detectCilia()* for more information about these data frames), as well as
- labeled images (converted to TIFs), which are saved in a subdirectory (called "output") of the directory with the image file (CZI/TIF).

Main steps of the main function detectCilia()

In essence, the following steps are executed:

1. Read the image data (`CZI` or `TIFs`) such that all z-stack layers will be in one 4-dimensional array (x, y, c, z, with c being the channel (red, green, blue)). ⟶ `Image_data` (type of all image data if not stated otherwise: formal class "Image" of package "EBImage").
2. If a `CZI` file is provided, read the metadata. ⟶ `df_metadata` (including `pixel_size` and `slice_distance`).
3. Calculate missing input parameter values. (For this, either `df_metadata` or specified `pixel_size` and `slice_distance` are needed.) ⟶ numerical values for `vicinity_combine`, `vicinity_connect`, `min_cilium_area_in_pixels`, `max_cilium_area_in_pixels`, and `nuc_mask_width_height_in_pixels`.
4. Calculate the maximum intensity projection of the z stack and, optionally, use the function *clahe()* (contrast limited adaptive histogram equalization) from the EBImage package to enhance the contrast of the image. ⟶ `Image_stack`.
5. Find and count all nuclei that do not touch the borders by applying several functions from the EBImage package as well as some cleaning measures. ⟶ `nmask_watershed` (type: formal class "Image" of package "EBImage").
6. Calculate the cilia detection thresholds, if not given, to find the cilia by using the ratio of cilia pixels R_{cp} by using `Image_projection` (see Equation 1). ⟶ `threshold_find` and `threshold_connect`.
7. Use `Image_projection` and the `threshold_find` to save a mask of possible cilia ⟶ `Image_cilia`.
8. Clean cilium pixels:
   - delete all found cilium pixels that have no neighboring (+-1) cilium pixels,
   - combine all cilium pixels that are no less than `vicinity_combine` apart,
   - delete all pixels that are not bright enough compared to other pixels of the same cilium,
   - delete all cilia structures that are too small or too large,
   - delete all cilia that touch a border of the image,
   - delete separated (by `vicinity_connect`) cilium parts that are not connected to the brightest cilium part.

   All remaining cilia are labeled in the z-stack projection. ⟶ `Image_projection_cilia` which is exported with the appendix "_stack_cilia_unconnected.tif".
9. Find in every z-stack layer the cilium pixels belonging to the cilia by using `threshold_connect` and the image `image_data` and clean the results applying the same cleaning methods as described above except for using `vicinity_connect` instead of `vicinity_combine`). ⟶ The pixels from every z-stack layer (i) belonging to a cilium are being labeled and every z-stack layer is exported with appendix "_cilia_layer_i.tif".
10. A data frame with all cilia information is being saved (z layer $= -1$ are the coordinates of the cilia from the projection and z layer $= -99$ are the cumulated cilia coordinates from the projection and every layer). ⟶ `df_cilium_information`.

11. A projection image with cilia pixels from the projection and every z-stack layer is saved. ⟶ `Image_stack_cilia_connected` which is exported with the appendix "_stack_cilia_connected.tif".
12. The labeled cilia and their ID (number) are added to the image and saved. ⟶ `Image_stack_numbers` which is exported with the appendix "_stack_cilia_all_numbers.tif".
13. The nuclei and labeled cilia and their ID (number) are added to the image and saved. ⟶ `Image_stack_numbers` which is exported with the appendix "_stack_cilia_all_numbers_nuclei.tif".
14. Normalized image with nuclei and with labeled cilia and their ID (number) are added and saved. ⟶ `Image_stack_histogram_equalization_normalized` which is exported with the appendix "_stack_cilia_all_histogram_equalized_normalized.tif".
15. Save function call and all used (default) parameters. ⟶ `df_parameterList` as well as CSV with "parameter_list.csv".
16. Save the number of nuclei. ⟶ "nuclei_number.csv".
17. Save information of cilia. ⟶ `df_cilium_summary` as well as "cilium_summary.csv".

Helper functions

The following helper functions are used by *detectCilia()*:

- `addNumberToImage.R`: Adds an image of a number to a specific position of another image.
- `editImage.R`: Takes a 3-dimensional array, a color channel, and a threshold value and returns a mask where a pixel value is 1 when the intensity is higher than the threshold and 0 otherwise.
- `getLayer.R`: Takes a 3-dimensional array (x, y, c) and a color channel and returns a 2-dimensional array (x, y), keeping only one required channel (red, green, or blue).
- `resizeImage.R`: Resizes an image with a given scaling factor (only used by `addNumberToImage.R`).
- `summarizeCiliaInformation.R`: Calculates the horizontal and vertical lengths of the cilia in µm.

Comparison of individual cilia

The measurement results of individual cilia of the seven test images are shown in Figures S4 and S5.

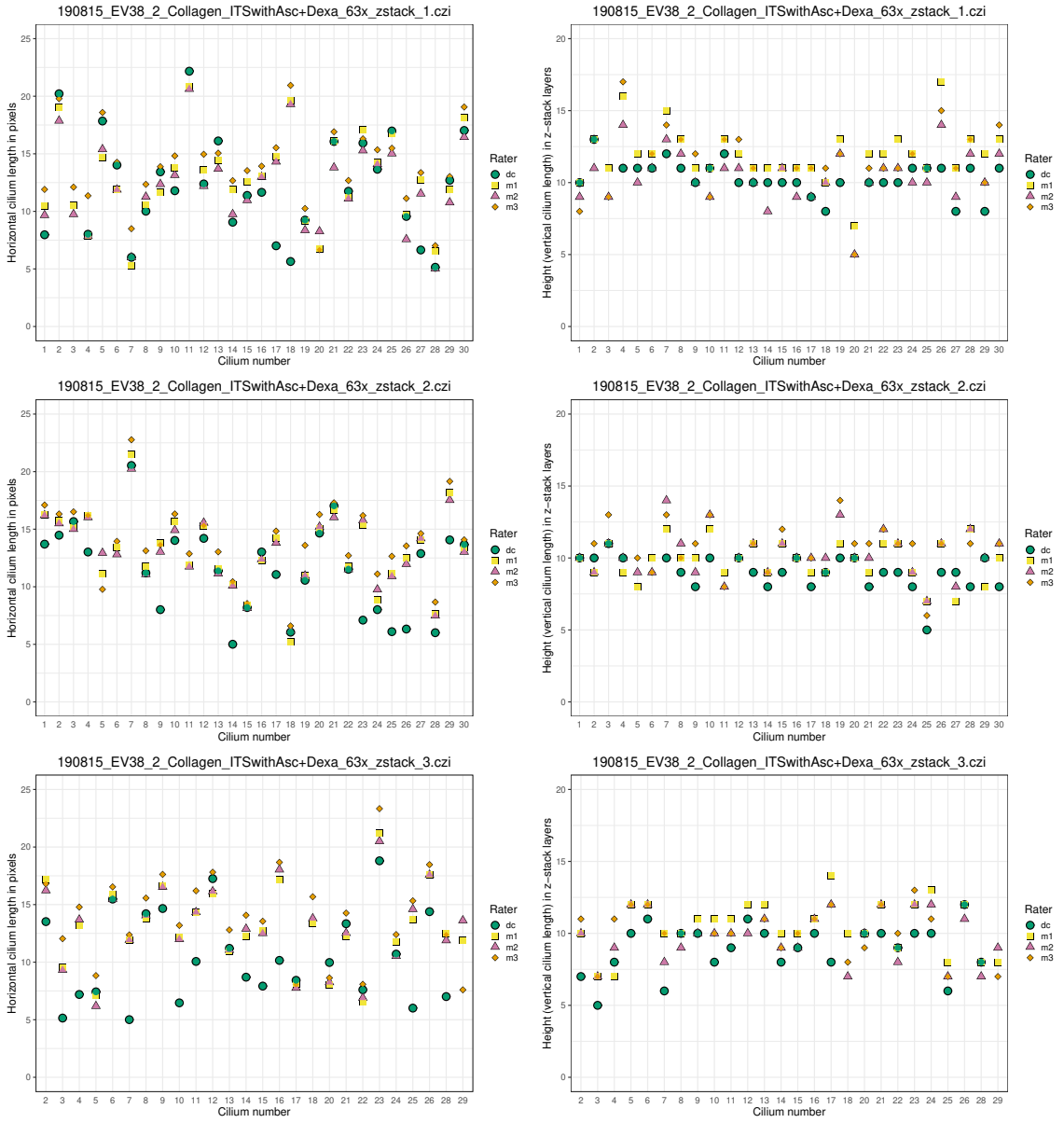

Figure S4. Results of the manual and automatic detection in Images 1–3 using detectCilia for every found cilium (horizontal and vertical lengths).

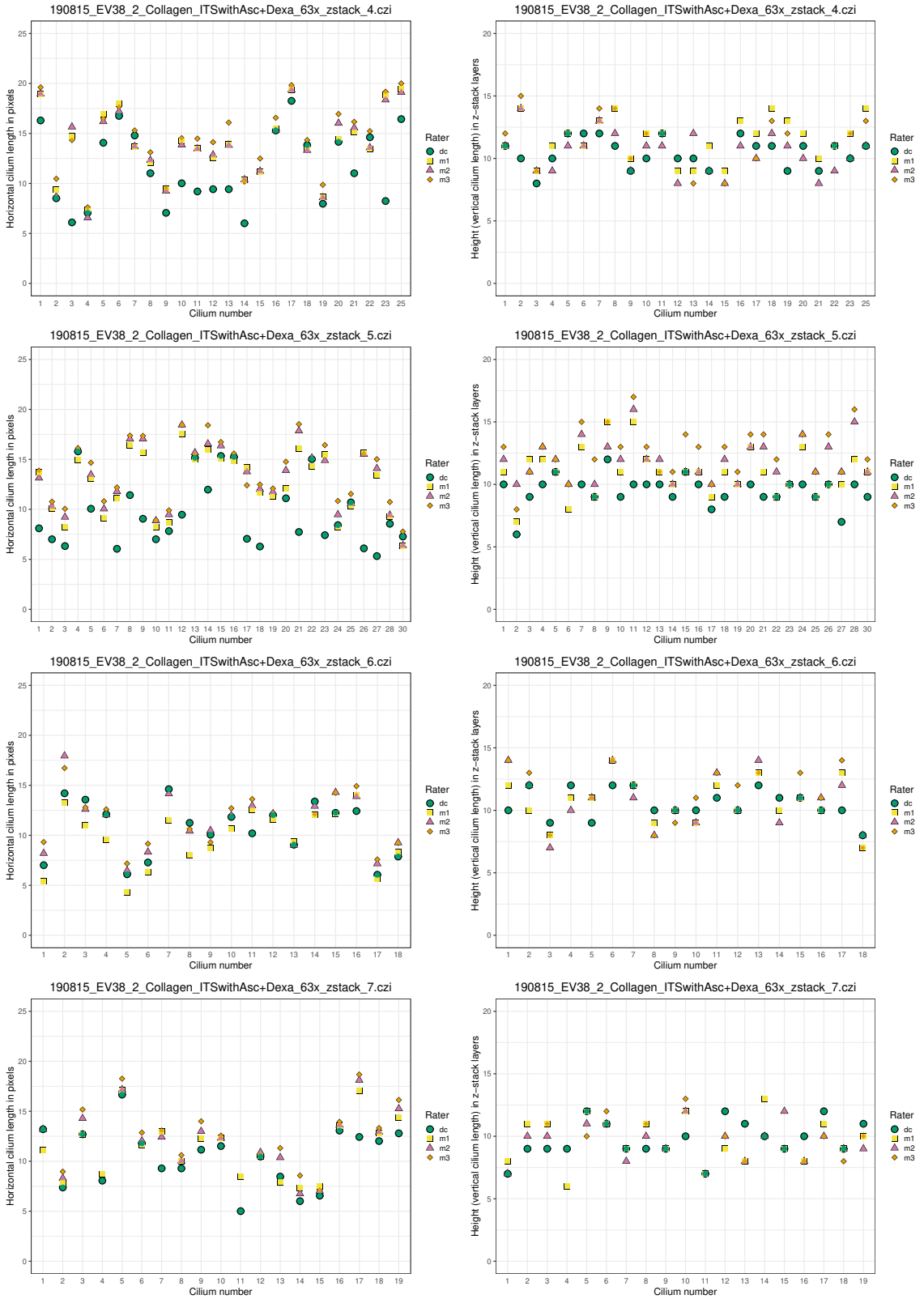

Figure S5. Results of the manual and automatic detection in Images 4–7 using detectCilia for every found cilium (horizontal and vertical lengths).

Impact of the pixel size on detectCilia results

We tested two different magnifications of the objective lens (63× and 100×) and three different frame sizes (1024^2, 2048^2, 4096^2) to find a suitable combination for measuring cilia. The influence of the settings on acquisition times and pixel sizes is shown in Table S5. The acquisition time t_{acq}

$$\begin{aligned} t_{\text{acq}} &\approx N_{\text{avg}} \cdot r^2 \cdot n_{\text{ch}} \cdot n_{\text{z}} \cdot t_{\text{pdt}} \\ &= 4 \cdot r^2 \cdot 3 \cdot 20 \cdot 1\,\mu\text{s} \end{aligned} \tag{3}$$

with the number of frame averages taken N_{avg}, the one-dimensional frame size r, the number of channels n_{ch}, and the number of z-stack layers n_{z} was estimated with a pixel dwell time t_{pdt} of 1 µs. The actual acquisition time was about two to three times higher than this estimated one. Table S5 also shows how many pixels represent a 3 µm long primary cilium and how much deviation an uncertainty of 2 pixels during detection would mean.

Table S5. Influence of the frame size and magnification on the uncertainty of the cilium length and acquisition time.

Frame size in pixels	Numerical Aperture	Pinhole diameter in Airy units (of cilium channel)	Objective lens magnification	Pixel size in µm	Number of pixels for a straight 3 µm cilium	Possible deviation for a detection error of 2 pixels in %	Acquisition time in minutes (according to Equation 3)
1024 × 1024	1.4	0.8	63×	0.220	14	14	4.2
1024 × 1024	1.46	0.6	100×	0.138	22	9	4.2
2048 × 2048	1.4	0.8	63×	0.110	27	7	16.8
4096 × 4096	1.4	0.8	63×	0.055	55	4	67.1

The comparison of our automated detection using detectCilia with a manual rater is shown in Figure S7. The results are in good agreement. Significant differences could only be seen for the highest digital resolution of Locations 3 and 4. The number of detected cilia was similar in the manual and automated detection groups after manually labeling false positive cilia (results not shown). The false positive rate was in Location 1: 31 % and 30 %, Location 2: 28 % and 21 %, Location 3: 35 % and 29 %, Location 4: 50 %, 56 %, and 42 %, and Location 5: % and 25 %. This high false positive rate was mainly due to a low signal-to-noise ratio, as seen in Figure S6, making detecting real cilia more difficult. The vertical and total lengths were not evaluated because the z stack did not cover the entire z range of the cilia.

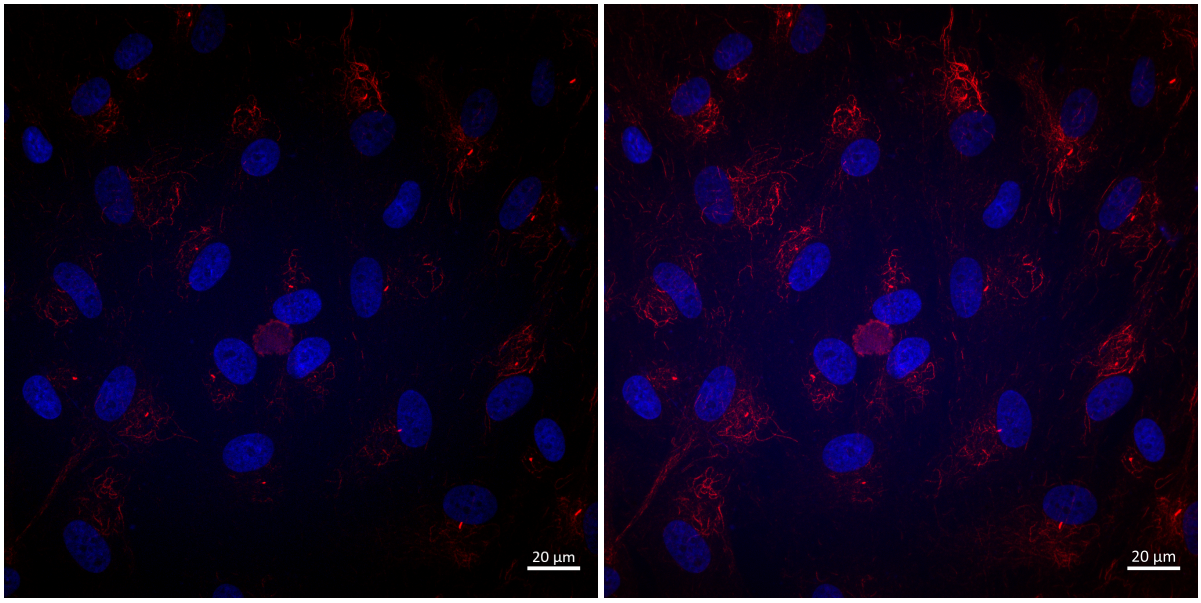

Figure S6. Z-stack projections of images of Location 3 (left: 1024 × 1024, right: 4096 × 4096) with identical brightness adjustments (blue channel max at 100/255, red channel max at 50/255). The higher frame size image was acquired before the other one, with a lower pixel integration time of 0.39 µs instead of 1.58 µs. (The green channel is not shown here.)

Although these results suggest that one should use small pixel sizes and z-step intervals when automatically detecting and measuring cilia, one should aim for fast image acquisition times and multiple images to get better summary statistics for the mean lengths, even if this

means that single values are less reliable. If we assume, on average, one cell per $900\,\mu m^2$, an image taken with a frame size of 1024×1024 and a pixel size of $0.220\,\mu m$ could contain 56 cells and cilia, and an image taken with a pixel size of $0.138\,\mu m$ could contain 22 cells and cilia. Thus, to avoid bleaching and get better summary statistics, we have used the $63\times$ 1.4 NA objective lens, a frame size of 1024×1024, and a zoom factor of 0.6, resulting in a pixel size of about $0.22\,\mu m$. Our z-step interval was $0.2814\,\mu m$.

In comparison, the authors of ACDC have used a $60\times$ 1.4 NA oil objective lens with a pixel size of $0.143\,\mu m$ and a $40\times$ 0.95 NA air objective lens with a pixel size of $0.1787\,\mu m$ for live cell imaging.[46] The authors of CiliaQ have used a $60\times$ objective lens (NA unknown) with a pixel size of $0.1036\,\mu m$ and a z-step interval of $0.5\,\mu m$ and a $20\times$ plan 0.8 NA objective lens with a pixel size of $0.1384\,\mu m$ and a z-step interval of $1\,\mu m$ for zebrafish embryos.[47]

The minor impact of the frame size/magnification on the quality of the results shows how robust our method is. However, one should keep in mind that the minimum frame size for an error of less than $5\,\%$, assuming ± 1 pixel error during detection, results in structure lengths of 40 pixels or for $<10\,\%$ error lengths of 20 pixels.

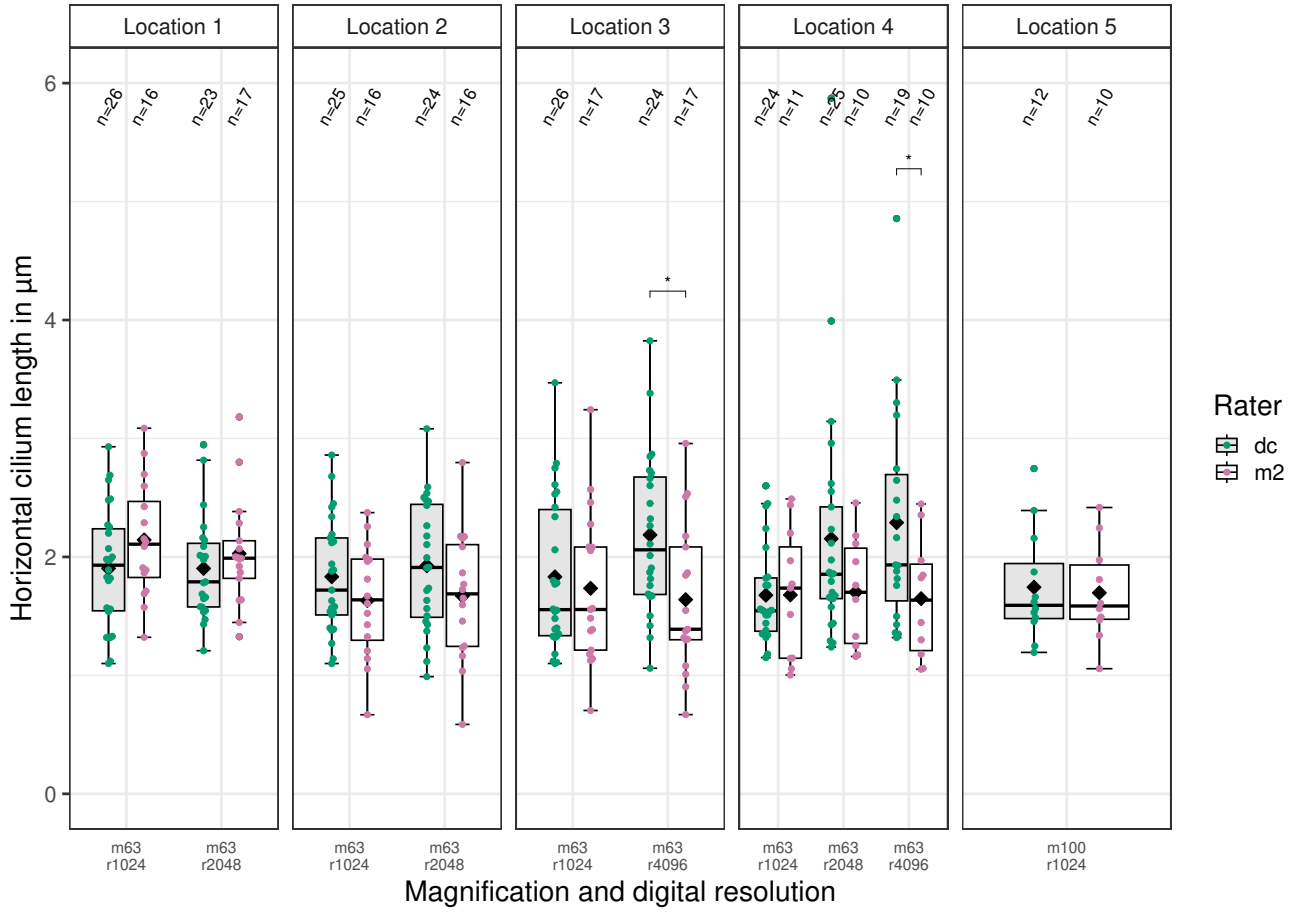

Figure S7. Number and horizontal lengths (in μm) measured in z-stack projections of detected primary cilia as measured manually or automatically by detectCilia (without manual correction) depending on the frame size and objective lens magnification.
Box-and-whisker plots with median, first and third quartiles, and outliers are shown. The black diamonds depict the mean values. The significance level is: *: $p < 0.05$. Non-significant comparison results are not shown.

Bending indices of cilia

CiliaQ calculates a bending index/curvature b for each cilium. It is "determined as the arc length along the cilium divided by the Euclidean distance of the first (cilium base) and last (cilium tip) skeleton point."[47] detectCilia, on the other hand, calculates a distance that is similar to the Euclidean distance mentioned above. Thus, the bending index b is a good approximation of the ratio of the cilia lengths determined by CiliaQ and detectCilia, or rearranged

$$l_{\mathrm{dc}} = l_{\mathrm{cq}}/b \tag{4}$$

with the cilia length determined by detectCilia l_{dc} and the cilia length determined by CiliaQ l_{cq} (cf. Equation 7).

Table S6 and Figure S8 show the results of CiliaQ for the 2D z-stack projection images and the original 3D data of the case study. It becomes clear that little cilia bending was detected in the 2D images. In contrast, in 3D, the cilia bending differed depending on the experimental group, and the cilia appeared about 1.5 to 1.8 times larger than a linear length measurement would indicate. This difference between 2D and 3D bending values means that CiliaQ has found the cilia to be more bent in the z direction than in the xy direction.

Table S6. Bending indices of cilia from z-stack projection (2D) and original (3D) images.

Group	2D Median	2D Mean	3D Median	3D Mean
ITS	1.17	1.21	1.71	2.05
ITS with Dexa	1.16	1.20	1.81	2.38
ITS with Dexa + IGF + TGF	1.16	1.21	1.52	1.90
FBS	1.13	1.17	1.68	1.96

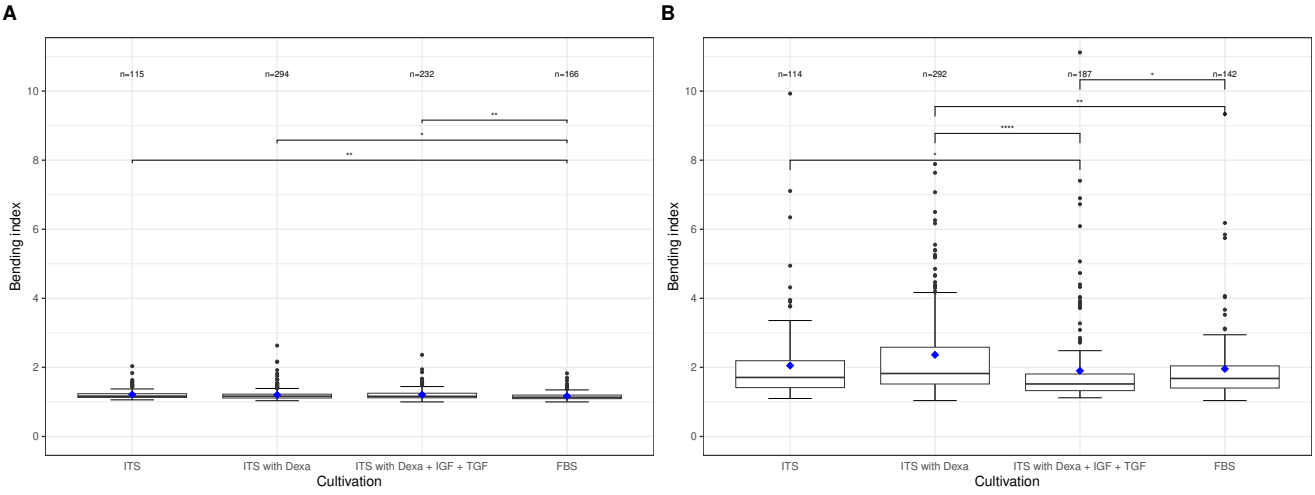

Figure S8. Bending index of cilia determined by CiliaQ. **A** depicts the index from z-stack projection images and **B** shows the index from the original 3D images. Box-and-whisker plots with median, first and third quartiles, and outliers are shown. The blue diamonds depict the mean values.

If a cilium were shaped like a circle with radius r and an opening described by an angle α (circular sector), then the bending index b would be $b = s/c$ with the arc length $s = r\alpha$ and the chord $c = 2r\sin(\alpha/2)$. This becomes

$$b = \frac{\alpha}{2\sin(\alpha/2)} \quad , \tag{5}$$

which is shown in Figure S9. Therefore, a sector with the central angle of $\alpha = \pi = 180°$ (semicircle) would lead to a bending index of $b \approx 1.6$, and a bending index of > 3 would be caused by an opening larger than $\alpha \approx 3\pi/2 = 270°$.

The length of a cilium L represented by such a circular sector measured by CiliaQ would be the arc length $L_{\mathrm{cq}} = s$ and measured by detectCilia

$$L_{\mathrm{dc}} = \begin{cases} 2r\sin(\alpha/2) & , 0 < \alpha \leq \pi \\ 2r & , \pi < \alpha < 2\pi \end{cases} \quad . \tag{6}$$

Therefore, the ratio is

$$\frac{L_{\mathrm{cq}}}{L_{\mathrm{dc}}} = \begin{cases} \frac{\alpha}{2\sin(\alpha/2)} = b \in [1, \pi/2] & , 0 < \alpha \leq \pi \\ \frac{\alpha}{2} \in (\pi/2, \pi] & , \pi < \alpha < 2\pi \end{cases} \quad . \tag{7}$$

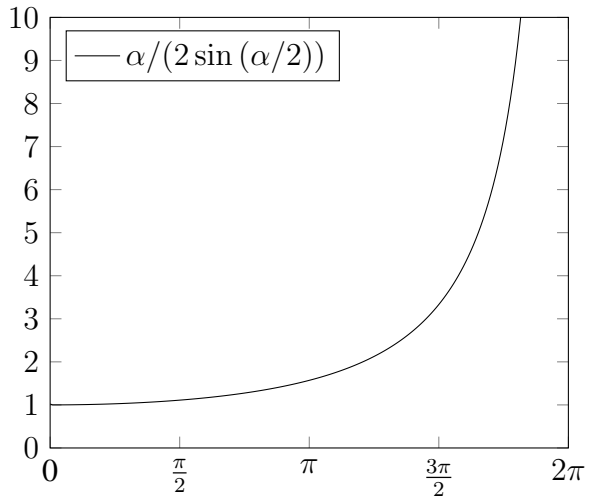

Figure S9. Bending index b of a circular sector with the central angle α.

Combined simulated cilia

Figure S10 shows the image containing all blurred cilia and the results obtained using detectCilia.

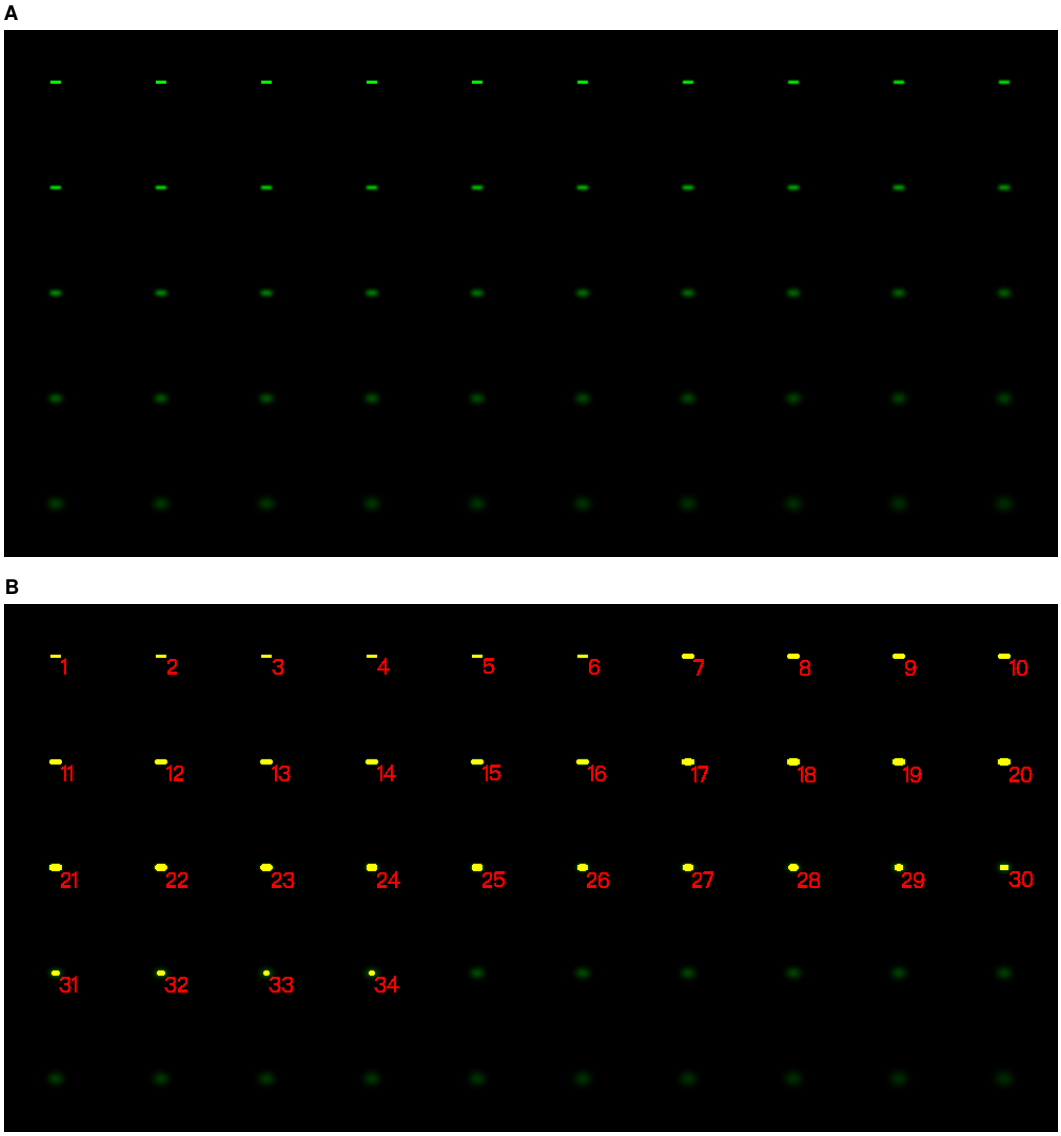

Figure S10. A shows the z-stack projection of all blurred cilia combined in one image used for cilia detection with all three tools, and **B** shows the result of detectCilia. detectCilia disregarded all cilia blurred with a Gaussian blur standard deviation larger than 3.3.

3D length measurement of a c-shaped simulated cilium

A c-shaped simulated cilium, as depicted in Figure S11, was used to test the 3D length measurements depending on the orientation of the cilium in 3D. The cilium covered 30 pixels (red and green squares in Figure S11A) in the xy-plane and had a height of 3 pixels (not shown). The skeletonized cilium (red pixels) leads to a total length (blue line) of $l = 2 \cdot \sqrt{2} \cdot 3 + 4 \approx 12.5$ pixels. The horizontal length (maximum distance from one to another cilium pixel) is $l_h = 10$ pixels. These values yield a bending index of $b \approx 12.5/10 = 1.25$. Before the detection, a Gaussian filter with a standard deviation of 1 pixel was applied to the image.

The result of the horizontal length measurement in the z-stack projection of the cilium rotated around the y-axis is shown in Figure S12A, and the total length measurement (in 3D) is seen in Figure S12B. The theoretical horizontal length l_h without Gaussian blur is given by the maximum of the z-stack projection lengths of the lengths in x, y, or z direction

$$l_h = \max\left\{c_l \cos\left(\alpha\right), 6, c_h \sin\left(\alpha\right)\right\} \quad , \tag{8}$$

with the cilium length $c_l = 10$ pixels and height $c_h = 3$ pixels. The horizontal length measurement without Gaussian blur deviates by a maximum of 2 pixels from the theoretical length. However, the theoretical values do not take into account that rounding is necessary when calculating the positions of voxels of the rotated object. Gaussian blur adds additional noise to the rotated image, which lets the horizontal length deviate up to 4 pixels from the theoretical values. Even though the length determination of detectCilia includes simplifications, it deviated less from the theoretical value than the measurement by CiliaQ for this specific simulated cilium.

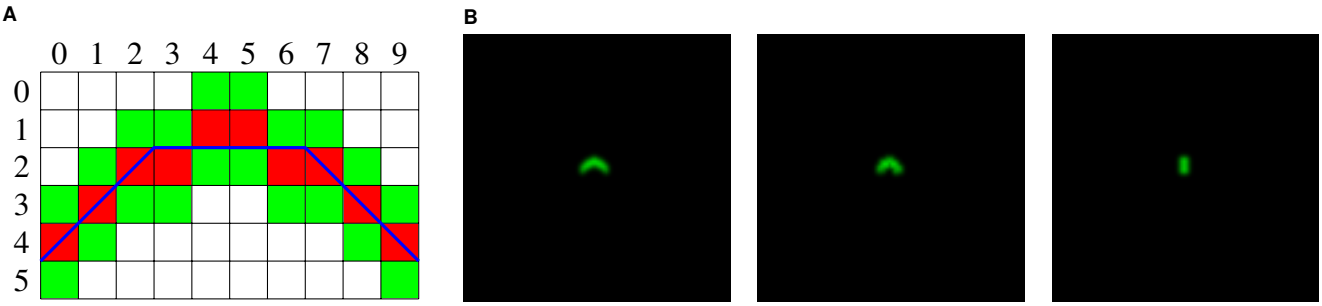

Figure S11. Simulated cilia c-shaped cilium with 30 pixels in the xy-plane and a height of 3 pixels (not shown). **A**: Positions of the cilium pixels in the xy-plane. The green and red squares resemble the cilium pixels. The red pixels alone depict the skeleton of the cilium. The blue line shows a possibility to measure the total length of the cilium. **B**: Images of the z-stack projections of the simulated c-shaped cilium rotated around the y-axis by $0°$, $45°$, and $90°$ with added Gaussian blur standard deviation of 1, respectively from left to right.

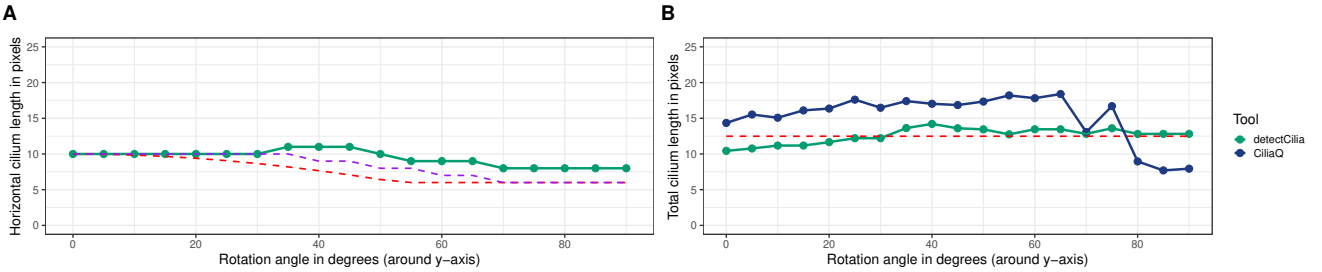

Figure S12. Length measurement of a rotated c-shaped simulated cilium depending on the rotation angle. **A:** Horizontal length with added Gaussian blur (green points) and without Gaussian blur (dashed purple line) as determined by detectCilia as well as the theoretical values (dashed red line). **B:** Total length as determined by detectCilia (in green) and CiliaQ (in blue) as well as the theoretical value (dashed red line).

Horizontal length of primary cilia in the case study determined by all tools

The differences in the horizontal lengths of primary cilia, as automatically determined by the three tools, detectCilia, ACDC, and CiliaQ, are shown in Figure S13 for the manually corrected data and in Figure S13 for the uncorrected, original results.

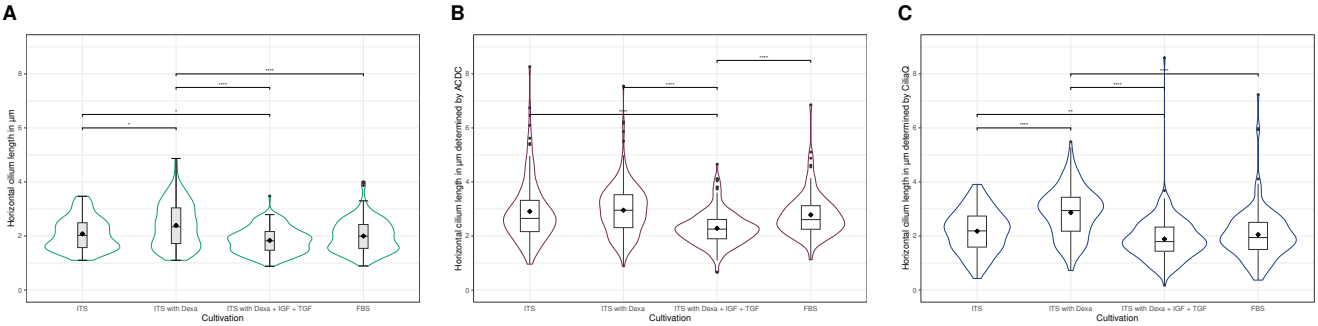

Figure S13. Comparison of the horizontal lengths of primary cilia using the manually corrected results of **A**: detectCilia, **B**: ACDC, and **C**: CiliaQ. Abbreviations: Dexa: Dexamethasone, FBS: Fetal bovine serum, IGF: Insulin-like growth factor, ITS: Insulin-transferrin-selenium, TGF: Transforming growth factor.

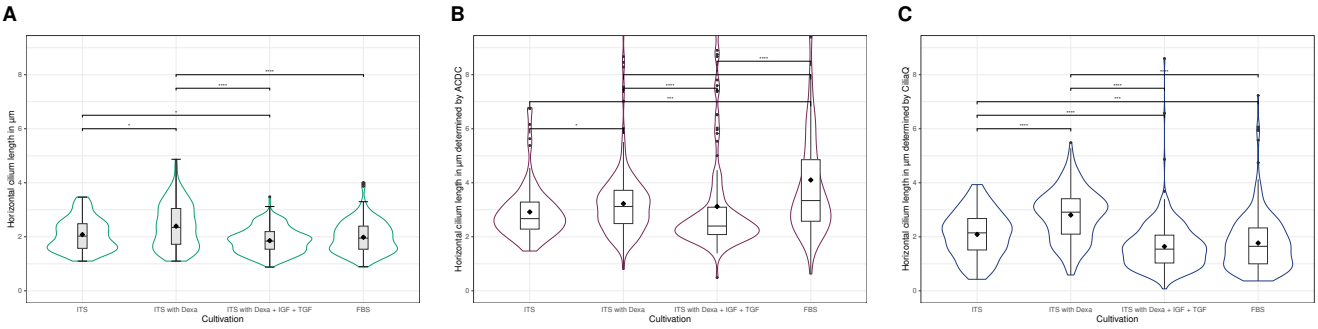

Figure S14. Comparison of the horizontal lengths of primary cilia using **A**: detectCilia, **B**: ACDC, and **C**: CiliaQ. The results were not manually corrected. Abbreviations: Dexa: Dexamethasone, FBS: Fetal bovine serum, IGF: Insulin-like growth factor, ITS: Insulin-transferrin-selenium, TGF: Transforming growth factor.