Science Advances
AAAS

advances.sciencemag.org/cgi/content/full/6/15/eaay7619/DC1

# Supplementary Materials for

## High-resolution 4-D acquisition of freely swimming human sperm cells without staining

Gili Dardikman-Yoffe, Simcha K. Mirsky, Itay Barnea, Natan T. Shaked*

*Corresponding author. Email: nshaked@tau.ac.il

Published 10 April 2020, *Sci. Adv.* **6**, eaay7619 (2020)
DOI: 10.1126/sciadv.aay7619

**The PDF file includes:**



**Other Supplementary Material for this manuscript includes the following:**

(available at advances.sciencemag.org/cgi/content/full/6/15/eaay7619/DC1)

## Supplementary Materials

### *Figures*

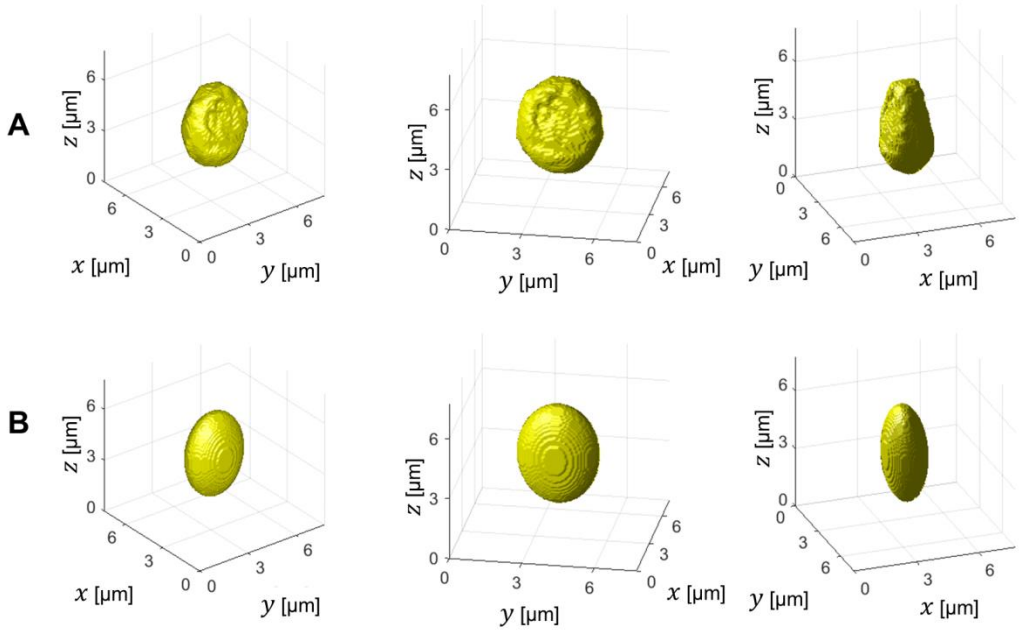


**Fig. S1. 3-D surface plot of an actual sperm cell (A) vs. its ellipsoid model (B), from various viewpoints.** The actual shape of the sperm cell is based on the results of the tomographic reconstruction obtained for experimental data (Fig. 2 in the main text), and coincides with the predicted shape of the head. Unlike the inner organelles of the sperm cell head, the 3-D shape of the sperm cell head surface is well known. The ellipsoid model in B was calculated based on the holograms of the same cell, as explained in Methods Section 3.8.

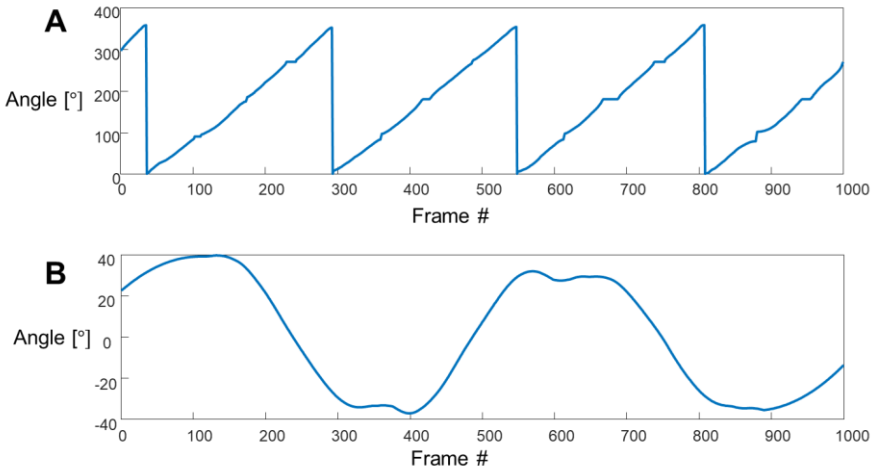


**Fig. S2. Experimentally recovered orientation of a human sperm cell during free swim.** (**A**) Roll angle. (**B**) Pitch angle.

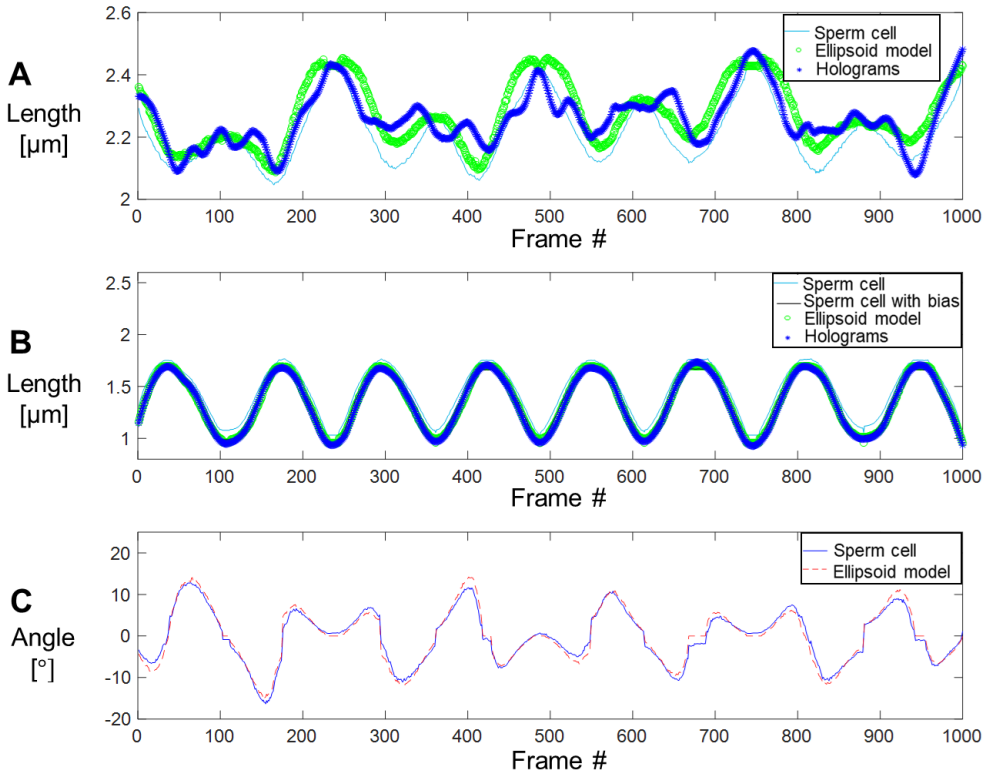


**Fig. S3. Comparison of various parameters calculated for the actual sperm cell head and the ellipsoid model, demonstrating negligible errors in calculating head orientation based on this model.** (**A**) Major radius. The root mean squared error (RMSE) is 0.069 μm between the results obtained from the sperm cell and ellipsoid model, and 0.07 μm between the result obtained from the sperm cell and the holograms. (**B**) Minor radius. The RMSE is 0.1 μm both between the results obtained from the sperm cell and the ellipsoid model, and between the results obtained from the sperm cell and the holograms. Here, we also show the result obtained for the sperm cell when adding a constant bias of 1.5 pixels to correct segmentation errors. After removing the constant bias, the RMSE is reduced to 0.03 μm between the result obtained from the sperm cell and the ellipsoid model, and 0.034 μm between the result obtained from the sperm cell and the holograms. (**C**) Yaw angle. The RMSE between the result obtained from the sperm cell and the ellipsoid model is 1.16°. The native yaw angle shown here cannot be reconstructed from the hologram itself, thus this third curve is not shown here.

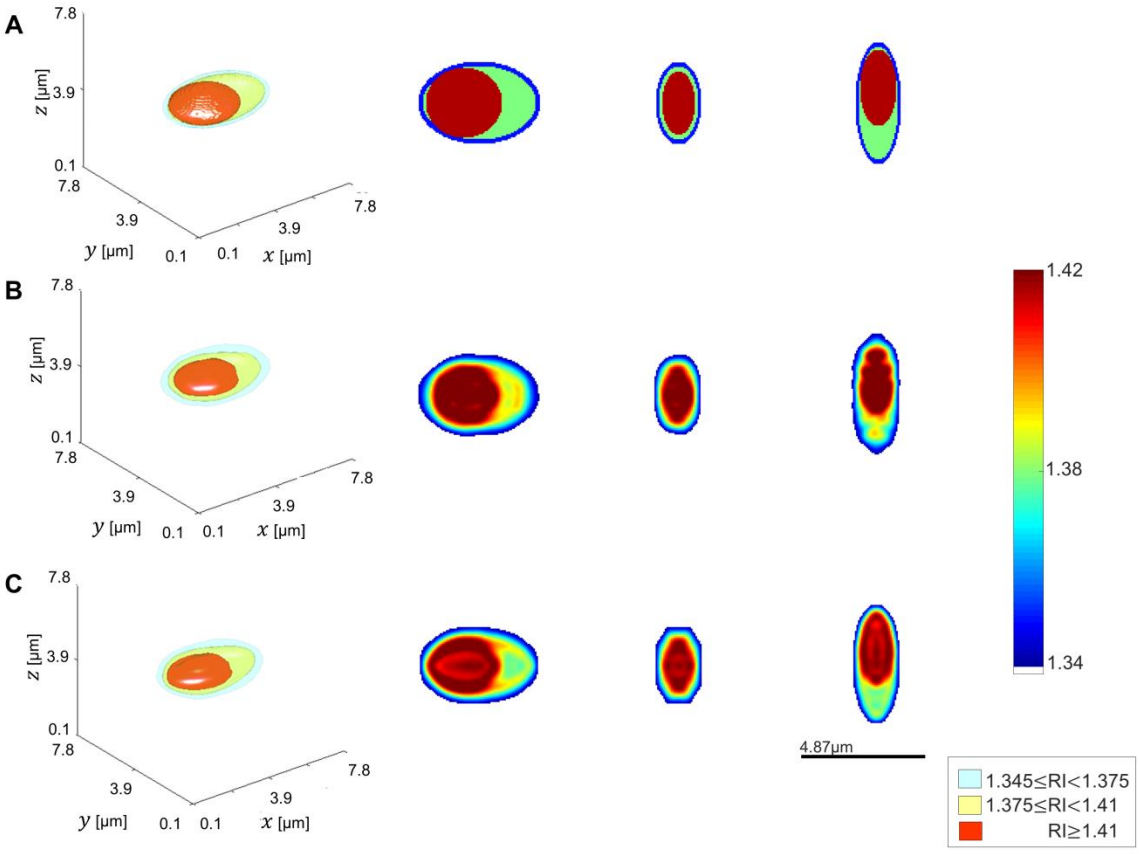


**Fig. S4. Simulation results verifying that the exact set of orientations obtained from the experimental data of the freely swimming sperm can obtain high-quality tomographic reconstruction.** (**A**) Ground-truth model of the RI distribution of a sperm head. (**B**) Reconstructed RI distribution obtained by using exactly the same orientations that have been available in the experimental case. (**C**) Reconstruction of RI distribution model using a single-axis 360° rotation at 5° angular increments, showing almost identical reconstruction quality as in B. In the leftmost column, the 3-D rendering is shown, with the corresponding legend provided in the bottom right corner. In the second column a mid x-y section is shown. In the third column a mid y-z section is shown. In the fourth column a mid x-z section is shown. The colorbar on the right displays RI values for the slice images.

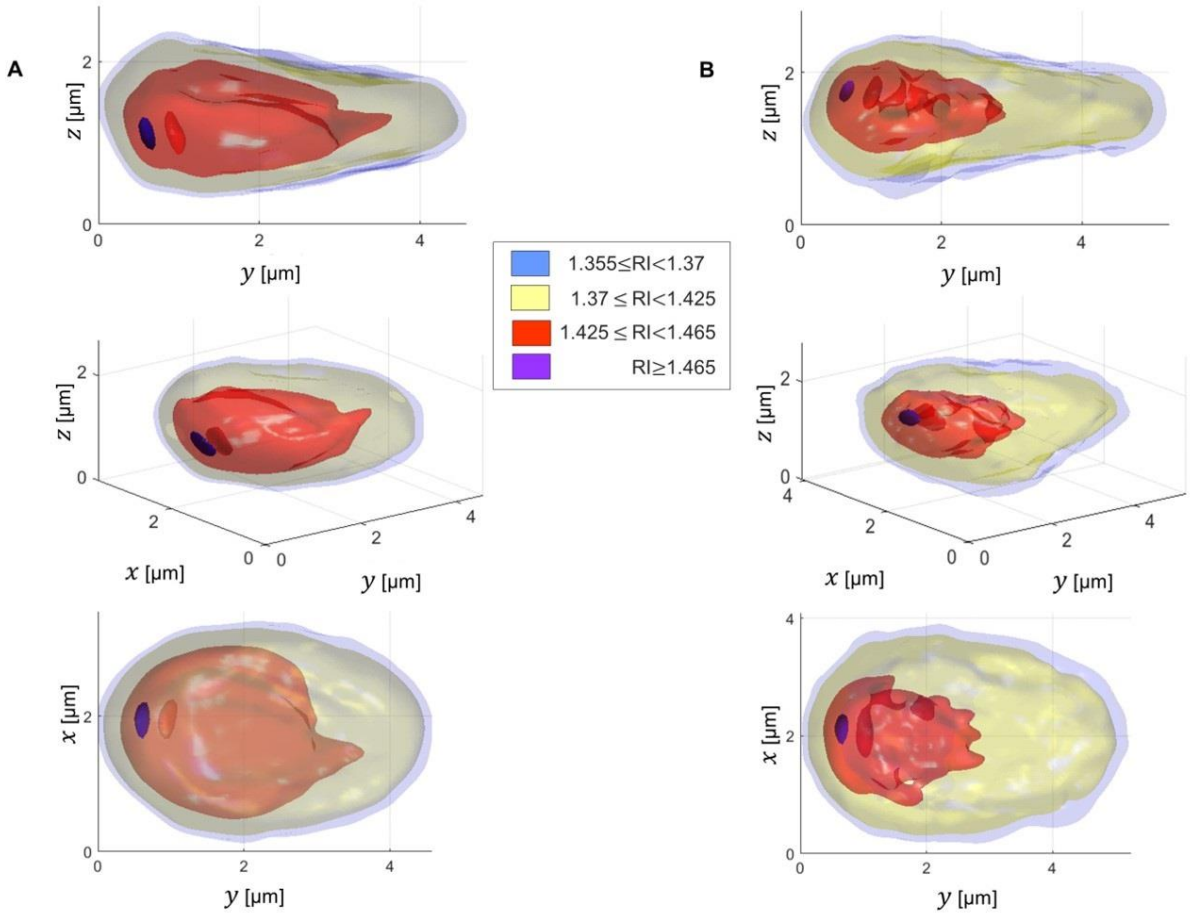


**Fig. S5. 3-D RI reconstructions of two additional sperm heads during free swim.** (**A**) Reconstruction based on 1000 frames, recorded at 2000 fps. (**B**) Reconstruction based on 500 frames, recorded at 1000 fps.

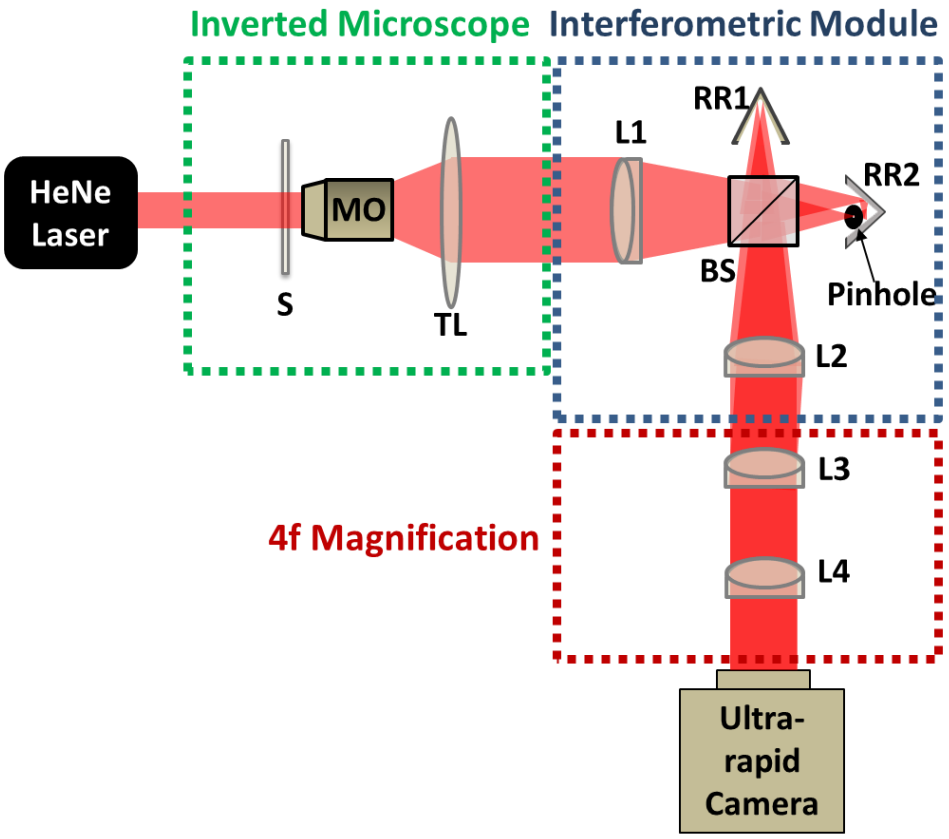


**Fig. S6. Diagram of optical system used to acquire the quantitative phase maps of the freely swimming sperm.** S, sample; MO, microscope objective; TL, tube lens; L1 - L4, achromatic lenses; BS, beam splitter; RR1, RR2, retro-reflector mirrors.

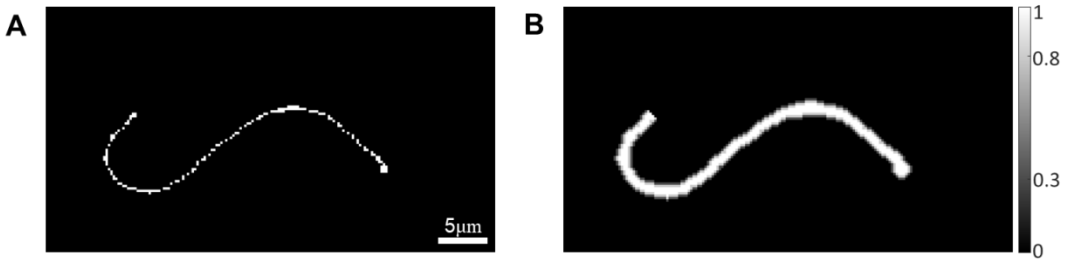


**Fig. S7. An example of the binary and weighting maps used for the recursive function (findFOC), for the frame shown in Fig. 3 in the main text.** (**A**) OLDMAP (**B**) OLDMAP2.

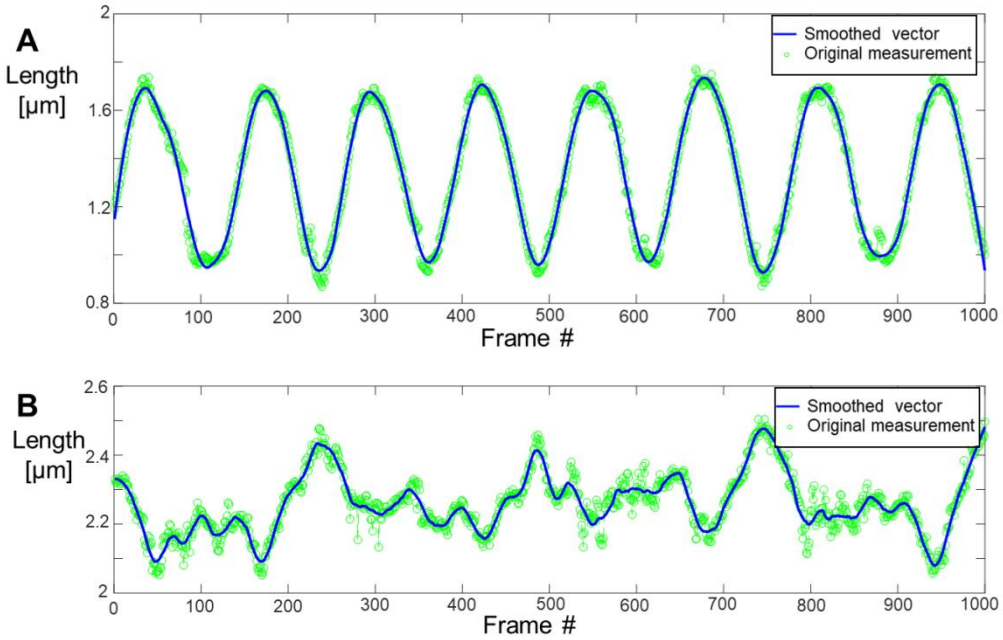


**Fig. S8. Major and minor radius measured for the experimental data, before and after smoothing.** (**A**) Minor radius. (**B**) Major radius.

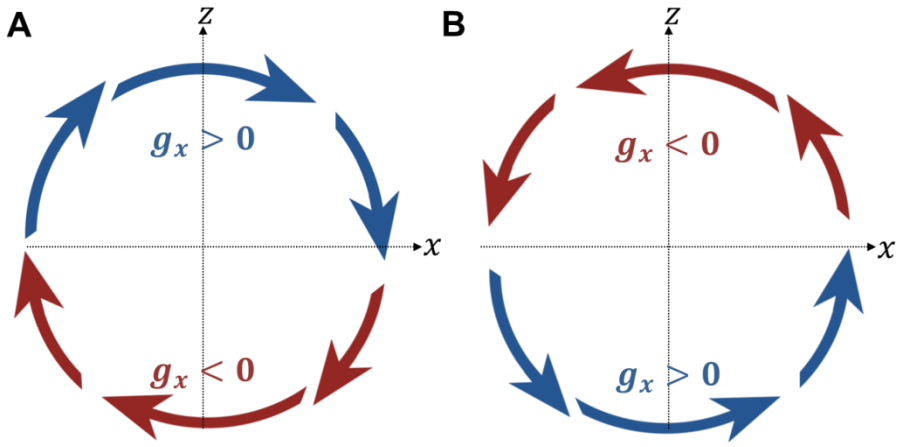


**Fig. S9. Method of finding the sperm head roll direction.** Red arrows indicate $g_x$ <0 , and blue arrows indicate $g_x$ >0. (**A**) Clockwise: higher values of z are obtained for the blue arrows ($g_x$ >0). (**B**) Counterclockwise: higher values of z are obtained for the red arrows ($g_x$ <0).

*Materials and Methods*

**Detailed algorithm (psudo-code) for processing the flagellum**

The recursive function, named findFOC, takes as input the coordinates of the last voxel that has been identified, the reconstructed complex field stack for that frame, six location maps (hit, hitc, hitz, OLDMAP, OLDMAPz, and OLDMAP2), an iteration counting parameter, the slope of the current segment, the noise level mode, and a validity flag variable initialized as a positive integer. The hit, hitc and hitz location maps store non-zero values in $(x,y)$ locations identified as being associated with the flagellum for the current frame, where the first stores consecutive integers that increase as the loop progresses (iteration count), the second stores "1"s, and the third stores the recovered depth for that location. OLDMAP is simply the hitc map for the previous frame, OLDMAPz is the hitz map for the previous frame, and OLDMAP2 is a dilated version of OLDMAP. OLDMAP2 is generated by applying multiple morphological dilations to OLDMAP with a disk-shaped structuring element 1 pixel in size, where pixels added by later dilations get a lower weight, yielding a weighting object that relies on smoothness (see Fig. S7) to indicate the likelihood of finding sperm pixels. In order to keep the endpoint in OLDMAP2 intact, we then zero out a 7×5 pixel environment around the row and column detected as associated with the end of the flagellum, defined as 3 pixels to each side, and 5 pixels forward, rotated at the direction of the slope of the last detected segment. For the first iteration of each frame, we use the coordinates of the seed point, the estimated slope at the neck ($S_n$) calculated in Section 3.3, and a neutral (i.e., 0) noise level mode. For the first frame processed, we use an altered version of the function (findFOC0) that does not require OLDMAP and OLDMAP2.

The findFOC function is composed of four parts:

1. **Initial estimation of current segment direction & $(x_i,y_i)$ coordinates of the next voxel associated with the flagellum.** In this part, we calculate the phase profile for the depth estimated from the previous voxel ($z_o$), and clean it by multiplying it by the weighting map OLDMAP2 (for findFOC0: using the TALCC function, with the parameters calculated from the previous iteration). Then, we estimate the slope at the previous location $(x_o,y_o)$; if the iteration counting parameter is lower than a fixed threshold, indicating that we are near the head, the slope is calculated based on thresholding the phase image to leave the head, and taking the orientation of the ellipse that has the same normalized second central moments as its binary mask. Otherwise, we crop the OLDMAP (for findFOC0: hitc) binary mask using a fixed window size centered on $(x_o,y_o)$, and estimate the orientation of the remaining object. This procedure only gives us the non-unique slope within a 180° range: -90°→90°; to find the slope within a 360° range, which is crucial for moving forward with the flagellum rather than backwards, we calculate the gradient of the hit location map (rotated according to the non-unique slope), the sign of which indicates the directionality of the search. To add to the reliability of the procedure, we assume continuity in the slope relative to the previous estimation, such that calculated slopes that are different by more than a fixed threshold from the previously calculated slope are discarded and replaced with the former result. This calculated slope, namely $S_N$, is used to define a 3×2 pixel environment centered on $(x_o,y_o)$, defined as 1 pixel to each side, and 1 pixel forward, rotated in the direction of the slope. Finally, we search the clean phase profile in the calculated environment for the maximal value, excluding pixels already flagged as visited by the hitc location map (which are flagged as "0"s), where the location of the maximal value yields the initial estimation of the $(x_i,y_i)$ coordinates of the next voxel associated with the flagellum. If all values in the environment are "0", indicating

that all locations in the environment have been visited, we define a validity flag of 0, indicating that we have reached the end of the flagellum for this frame (stopping criterion), and return to the calling function immediately.

2. **Estimation of the $z_N$ location of the $(x_i,y_i)$ coordinate.** In this part, we first find the possible range of depths based on continuity and smoothness considerations, both relative to the current and previous frame (meaning in 4-D); towards this end, we first extract the depth range from a 5×5 pixel environment in the OLDMAPz map, defined as 2 pixels to each side, and 4 pixels forward, rotated in the direction of the slope, starting at $(x_i,y_i)$, and calculate its median, $z_m$. We then use this information as follows:

    - If $z_m < z_o$, we take $[z_o -2, z_o]$.
    - If $z_m > z_o$,we take $[z_o, z_o +2]$.
    - If $z_m = z_o$ (or there is insufficient information in the area defined by the 5X5 pixel environment in the OLDMAPz map), we simply take $[z_o -2, z_o +2]$.

Next, we examine a linear section of the TALCC-cleaned phase image centered on $(x_i,y_i)$ which is orthogonal to the direction of the current segment (based on $S_N$), for the calculated depth range. Since the midpiece area is characterized by a thicker segment, such that it can endure more rigorous cleaning on the one hand, and is less sensitive to our noise detection criteria on the other hand, by default we use a higher noise level mode when cleaning it with the TALCC function (see Section 3.5). For every depth in the calculated range, we calculate the standard deviation and a score expressing the likelihood of this depth being the location of optimal focus, based on the following parameters:

1- Weighting according to distance from the center of the estimated depth range. Assuming continuity and smoothness, it is most likely that the depth of the following pixel will be identical to the current one, slightly less likely for it to differ by ±1, and increasingly less likely as the difference increases.

2- The width of the section (number of pixels with value above threshold).

3- The amplitude of the section (value of highest peak).

To minimize the effect of local noise, we smoothed the width and amplitude vectors with a window size equal to the vector length. Next, the distance-weighting, width, and amplitude-weighting vectors obtained for the entire depth range are used to calculate a triple-score-vector, given by the element-wise multiplication of the distance-weighting and width vector, divided (element-wise) by the amplitude vector. At this point, we need to check the validity of the score, which is linked to the noise level of the image. If the image is noisy even after the TALCC function, the width parameter would be unreliable, as it may include background pixels. On the other hand, if the cleaning procedure was too aggressive, we may have excluded pixels that should have been counted. Thus, the score is flagged by replacing it with a large constant number in the following cases:

- The amplitude value is 0.
- The segment has no maximum points (shape is wrong).
- The width of the segment is higher than a threshold (set to a different value in the thicker midpiece area than in the narrow flagellum area).
- The width of the segment is two pixels or less, and the width of the respective segments in adjacent depths differs by over two pixels (suggesting a mistake in segmentation).

- An additional condition only relevant for non-midpiece area: the segment has minimum points (i.e., the shape is wrong).

Afterwards, the closest neighbors of the depths that received the large constant number have their scores doubled as a penalty, with the assumption being that it is less likely for a depth adjacent of these invalid depths to be the ideal focus plane.

Then, the depth with the lowest score – thus with the steepest, most narrow segment (that did not fall into any of the categories above) – corresponds to the ideal focus plane, yielding the depth of the current location, $z_N$. If the value of the lowest score is the high constant number used as a flag, we infer that the noise level is still too high and return a "0" noise flag to the calling function. We also return a "0" noise flag (indicating high noise levels) to the calling function if we are out of the midpiece area with a noise level mode lower than three (meaning more rigorous cleaning was not applied) and there is either a jump of more than two pixels in the width vector (indicating unrelated pixels attributed to the count), or more than one depth with an invalid shape.

Finally, we calculate two additional global parameters to return to the calling function, indicating either excessive cleaning or that the end of the flagellum was reached:

1- The mAMP parameter, storing the maximal value of the amplitude vector.

2- The mSTD parameter, storing the maximal value of the standard deviation vector (calculated for the raw, uncleaned phase image).

**3. Validity check of the result and recursion**

In this part, we look at the output of the previous part and decide whether the result is valid, more/less cleaning is needed, or we have reached the end of the flagellum.

- If the current noise level mode is non-negative (indicating neutral or high noise) but smaller than 5 (to prevent an infinite loop) and the noise flag from the previous step is 0: we recall the findFOC function with a noise level mode increased by 1, with all the initial parameters, including $(x_o,y_o,z_0)$.
- If the current noise level is non-positive (indicating neutral noise or delicate segments) but larger than −3 (to prevent an infinite loop), and mAMP or mSTD are lower than a fixed threshold: we recall the findFOC function with a noise level mode decreased by 1, with all the initial parameters, including $(x_o,y_o,z_0)$.
- If none of the above conditions were met and mAMP or mSTD are lower than a fixed threshold, we decrease the value of the validity flag by 1.

4. **Final estimation of the current segment direction and the $(x_N,y_N)$ coordinates of the next voxel associated with the flagellum.** In this part, executed only once for the recursion base, with the updated noise level mode value, we repeat step 1 using the updated depth $z_N$, estimated at Step 2 above. If the phase value of $(x_N,y_N,z_N)$ is below a fixed threshold, we decrease the value of the validity flag by 1 (unless it has already been decreased by 1 in this iteration).

For each frame, we iterate until we reach a value of 0 for the validity flag, indicating that we have reached the end of the flagellum for this frame (stopping criterion). The tolerance for local errors vs. the ability to accurately pinpoint the end-location of the flagellum is a result of the positive integer chosen for the initialization of the validity flag variable. If the flagellum detection has ended abruptly by reaching a validity flag of 0, we check the Euclidean distance between the current and previous end-point; if it is larger than a fixed threshold, we assume an error has occurred in this specific

frame (as may occur due to high local noise), define this frame as invalid, and use the previous frame again instead, slightly increasing the distance tolerance for the next frame.

**Movie S1**. **Focusing algorithm in action**. The red vertical segment moves along the flagellum from the neck to the distal end, finding the ultimate focus plane for each location it is in.

**Movie S2. 2-D representation of the 3-D segmentation map of the flagellum.** The colormap indicates the recovered depth (relative to the original focus plane).

**Movie S3. Full 3-D motion reconstruction of a human sperm cell swimming freely.**

**Movie S4. Bright-field imaging of sperm cells with no added motion altering substances, in HTF with 7% PVP, in HTF only, and in HTF with 5 mM caffeine.**

**Movie S5. Full 3-D motion comparison for two human sperm cells from the same donor, with and without caffeine.**