www.sciencetranslationalmedicine.org/cgi/content/full/7/299/299ra124/DC1

Supplementary Materials for

Automated identification of abnormal respiratory ciliary motion in nasal biopsies

Shannon P. Quinn, Maliha J. Zahid, John R. Durkin, Richard J. Francis, Cecilia W. Lo,*
S. Chakra Chennubhotla*

*Corresponding author. E-mail: cel36@pitt.edu (C.W.L.); chakracs@pitt.edu (S.C.C.)

Published 5 August 2015, *Sci. Transl. Med.* **7**, 299ra124 (2015)
DOI: 10.1126/scitranslmed.aaa1233

This PDF file includes:

Materials and Methods
Fig. S1. Aggregate optical flow displacement in CHP data cohort.
Fig. S2. Pixel selection in an ROI.
Fig. S3. Breakdown of digital nasal biopsy video data sets.
Fig. S4. CM classification pipeline.
Fig. S5. Classification confidence as a function of ROIs per patient.
Fig. S6. Web site proof-of-concept screenshots.
Fig. S7. Pairwise angles between principal components of CM in AR models.
Fig. S8. CM classification results of parameter scanning.
Table S1. Constant parameters used throughout this study.
Legends for movies S1 to S9
Reference (*42*)

Other Supplementary Material for this manuscript includes the following:
(available at
www.sciencetranslationalmedicine.org/cgi/content/full/7/299/299ra123/DC1)

Movie S1 (.mov format). Example of normal CM of nasal biopsy from control.
Movie S2 (.mov format). Example of abnormal CM of nasal biopsy from PCD patient.
Movie S3 (.mov format). Example of abnormal asynchronous and wavy CM.
Movie S4 (.mov format). Example of abnormal CM with incomplete stroke.
Movie S5 (.mov format). Example of abnormal CM with asynchronous beat and incomplete stroke.

Movie S6 (.mov format). Example of a video capture artifact of extraneous tissue motion.
Movie S7 (.mov format). Example of a video capture artifact of poor camera focus.
Movie S8 (.mov format). Example of a false-negative prediction.
Movie S9 (.mov format). Example of a false-positive prediction.

Materials and Methods

Image velocities and elemental motions

In this section we motivate the use of differential features of optical flow for characterizing ciliary motion dynamics. First, we provide an overview of optical flow research for general scientific audiences. The subsequent discussion follows the notation from Kooenderink *et al* (30) and Kovesi *et al* (42), in which we provide the optical flow and image velocity derivations.

Overview

Optical flow is apparent motion of objects in a scene caused by the relative motion between an observer and the subject. Optical flow can be used for tracking objects, but tracking across more than each pair of frames is not an inherent property of optical flow, and is not the strategy we use in this work. Instead, we work directly with the vector fields generated from each pair of frames in the original video (Fig. 1C,D).

For each pixel at which we compute flow, constraints on the direction and magnitude are incorporated by considering a neighborhood around the pixel. While the size of the neighborhood and strength of contribution of the neighborhood to the final optical flow of the pixel of interest varies across implementations, pooling these constraints over a neighborhood accounts for and eliminates a certain quantity of noise. This also results in the overdetermined system of equations that is referenced in the following section.

There are a list of “best practices” when computing optical flow, such as temporal smoothing and median filtering, which we adhere to (Materials and Methods, Sun *et al* (29)). Additional instances of smoothing and filtering of noise in this work include spatial median filtering of dominant frequencies for frequency histograms, and the principal component analysis in the AR models which discards all but the dominant modes of motion. Since the publication of the Canny edge detector in 1986, all derivatives in image processing are preceded by Gaussian filtering with an appropriate scale parameter. We do the same in computing optical flow and its derivatives of rotation and deformation. The effect of this filter can be observed in Fig. 2E-G: there are no abrupt changes in the waveform, and the waveforms remain roughly periodic where appropriate.

We chose the Farneback optical flow implementation in OpenCV over the Sun *et al* (29) version in Matlab out of practicality. Both algorithms produced near-identical optical flow fields; however, the

Sun algorithm incorporated non-local optimizations for constraining the optical flow computations over much larger neighborhoods. While this could help the reliability of optical flow computations in videos with large discontinuities, e.g. shaking cameras or sudden rapid movements, we found this was not a particularly prevalent issue in our study; fig. S2 shows that over 98% of the observed motion displacement occurred over fewer than 1 pixel; over 99% of the displacement spanned fewer than 2 pixels. Given the relatively small displacement across our data, and that the non-local considerations in the Sun *et al* algorithm resulted in significantly longer optical flow computations, we elected to use the Farneback OpenCV algorithm. Furthermore, by using the OpenCV algorithm, we could implement our entire framework in Python.

Optical flow has also been used extensively in dynamic texture recognition in the form of rotation, divergence, and deformation, as first explored by Koenderink *et al* (31). Extensive work has been done applying optical flow to the recognition of texture waveforms in dynamic video data (22-28). In particular, (27) applies this technique to the biomedical and clinical contexts, analogous to our work.

There is also some precedent in the literature for using image processing in the analysis of ciliary motion. In one work (13), the authors used the OpenCV framework for reading the video data, as we did. The authors then computed local gradients of the video frames to estimate CBF; no particle tracking was performed, though these local gradients would be the first step to computing optical flow. The second work (21) was our own, in which we used optical flow and its derivatives to compute preliminary results upon which this work capitalizes. We built rotation and deformation magnitude histograms for a handful of clinically recognized ciliary motion types.

Ultimately, optical flow methods have been extensively studied and compared with ground truth in the computer vision community. We cite several landmark studies in this regard; optical flow has even been used in Hollywood for special effects (see Michael Black's talk [1]). However, there is no ground truth for optical flow of ciliary motion; we cannot generate error bars for optical flow of ciliary motion as the current ground truth of ciliary beat pattern analysis is manual expert assessment. The closest comparison we can make is to correlate our classification results with these expert assessments. A future experiment for establishing ground truth of optical flow in ciliary motion could entail comparing the computed flow of beads among the cilia biopsies to manual tracking of the beads' motions.

[1] `https://www.youtube.com/watch?v=tIwpDuqJqcE`

Optical Flow

Optical flow computation follows from invoking the standard *brightness constancy assumption*,

$$I(x, y, t) = I(x + u\delta t, y + v\delta t, t + \delta t), \tag{1}$$

which states that image intensity I (or a filtered version of it) at a location (x, y) at time t is preserved locally for small changes $(u\delta t, v\delta t)$ observed in space in a small time interval δt. Here (u, v) are the horizontal and vertical image velocity components of the optical flow vector $\vec{f}^{\,\mathrm{T}}$ at pixel location (x, y).

A first-order approximation of the right hand term in the brightness constant equation above gives rise to the gradient constraint:

$$I_x u + I_y v + I_t = 0, \tag{2}$$

where the subscripts x, y and t on image intensity I denote partial derivatives of the image structure at location (x, y). The gradient constraint is pooled over a small image neighborhood around pixel (x, y) to form an overdetermined system of linear equations from which the optical flow vector (u, v) is estimated. We use a variation of the classical optical flow algorithm suggested by Black and co-workers that incorporates a non-local smoothness term to integrate information over larger neighborhoods. While optical flow vector $\vec{f} = (u, v)^{\mathrm{T}}$ provides information on the image dynamics, the first-order flow derivatives: (u_x, u_y, v_x, v_y), can be additionally used to derive a linear (affine) model of optical flow, and provide a statistical means of characterizing ciliary dynamics.

Differential Invariants

Consider two spatially nearby image points $\vec{r}_1 = \vec{r}$ and $\vec{r}_2 = \vec{r} + \delta\vec{r}$ along a cilium. The vector $\delta\vec{r} = \vec{r}_2 - \vec{r}_1$ gives their relative position. We assume that the points move according to their optical flow velocities $\vec{f}_1 = \vec{f} = (u, v)^{\mathrm{T}}$ and $\vec{f}_2 = \vec{f} + \delta\vec{f}$ and after a small time interval δt they are at locations $\vec{r}_1{}' = \vec{r}_1 + \vec{f}_1 \delta t$ and $\vec{r}_2{}' = \vec{r}_2 + \vec{f}_2 \delta t$. It follows that:

$$\vec{r}_2\,' - \vec{r}_1\,' = (\vec{r}_2 - \vec{r}_1) + \left(\vec{f}_2 - \vec{f}_1\right)\delta t, \tag{3}$$

$$\delta\vec{r}\,' = \delta\vec{r} + \delta\vec{f}\delta t. \tag{4}$$

Given the spatial nearness of the two points $\vec{r}_1$ and $\vec{r}_2$, we can relate the flow vectors $\vec{f}_1$ and $\vec{f}_2$ by Taylor series expansion that uses first-order differentials of optical flow:

$$\vec{f}_2 \approx \vec{f}_1 + \frac{\partial\vec{f}_1}{\partial\vec{r}}\delta\vec{r} + \cdots , \tag{5}$$

$$\vec{f}_2 \approx \vec{f}_1 + \begin{pmatrix} u_x & u_y \\ v_x & v_y \end{pmatrix} \delta\vec{r} + \cdots , \tag{6}$$

where (u_x, u_y, v_x, v_y) are elements of the spatial derivative of optical flow, i.e. flow gradient: $\frac{\partial\vec{f}}{\partial\vec{r}}$. As shown by Kooendernik and van Doorn, the flow gradient can be further decomposed into scaling (divergence), shearing (deformation) and rotational (curl) components. These are scalar quantities defined as:

$$\mathrm{div}\vec{f} = u_x + v_y, \tag{7}$$

$$\mathrm{rot}\vec{f} = v_x - u_y, \tag{8}$$

$$\left(\mathrm{def}\vec{f}\right)\cos(2\mu) = u_x - v_y, \tag{9}$$

$$\left(\mathrm{def}\vec{f}\right)\sin(2\mu) = u_y + v_x, \tag{10}$$

where μ is the angle of maximal distortion (described below). The quadruplet of quantities:

$$\mathrm{div}\vec{f}, \mathrm{rot}\vec{f}, \left(\mathrm{def}\vec{f}\right)\cos(2\mu), \left(\mathrm{def}\vec{f}\right)\sin(2\mu)$$

form a linear space and provide an equivalent representation of flow gradient $\frac{\partial\vec{f}}{\partial\vec{r}}$. Observe that the deformation magnitude and orientation can be derived as:

$$\mathrm{def}\vec{f} = \sqrt{(u_x - v_y)^2 + (u_y + v_x)^2}, \tag{11}$$

$$2\mu = \arctan\left(\frac{u_y + v_x}{u_x - v_y}\right). \tag{12}$$

The quantities def$\vec{f}$, div$\vec{f}$, and rot$\vec{f}$ are differential invariants as they are independent of coordinate system used to measure the flow.

Using these definitions the velocity gradient can be rewritten as:

$$\begin{pmatrix} u_x & u_y \\ v_x & v_y \end{pmatrix} = \frac{\mathrm{div}\vec{f}}{2}\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \frac{\mathrm{curl}\vec{f}}{2}\begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} + \tag{13}$$
$$\frac{\mathrm{def}\vec{f}}{2}\begin{pmatrix} \cos(2\mu) & \sin(2\mu) \\ \sin(2\mu) & -\cos(2\mu) \end{pmatrix}.$$

Figure 1E illustrates the geometric/image distortions that each of these differential features are associated with; these are discussed in greater detail below.

Divergence

Divergence is image distortion seen geometrically as a local isotropic expansion with speed $\frac{1}{2}\,\mathrm{div}\vec{f}$ about a focus of expansion. We do not expect these distortions to appear in the lateral views of the cilia, but they could be useful in characterizing the ciliary motions captured by a perpendicular view of the cilia. Divergence is invariant to the orientation of the cilia in the image plane.

Rotation

The most salient feature of ciliary motions is the sweeping forward and backward strokes. Rotation (or curl) captures the local rotation of cilia with angular velocity $\frac{1}{2}\,\mathrm{rot}\vec{f}$. Note this rotation is a component perpendicular to the viewing direction. Curl is orthogonal to divergence. Like divergence, curl is invariant to the orientation of the cilia in the image plane.

Deformation

Deformation measures distortions that affect orientation of a ciliary region while preserving apparent areas. Two axes, the axis of maximal extension and the axis of maximal contraction, form an orthogonal basis for describing all possible motion field distortions or shearing motions. Since cilia are stuck to the cell wall, it is more appropriate to see their motions as having both a rotational component and a shearing motion (hence a directed shear).

Differential Feature Filters

Ciliary motion videos have high sampling frequency (200Hz) relative to their natural beats ($\approx$ 10Hz). While it is easy to construct the optical flow derivatives with Gaussian derivatives, it is also instructive to consider how to design a filter mask that can elicit differential flow information. Intuitively, the detector template mask would resemble a miniature vector field exhibiting *atomic* motion types: divergence, curl or deformation. The magnitudes of individual vectors in the motion template will be proportional to distances from the center of the filter and the vector directions will be a function of atomic motion type. Indeed, Eqns. 10-13 make it obvious how to construct these filters, as shown in Fig. 1E.

The divergence filter mask has vectors that point radially outward, whereas the curl filter has tangential vectors. Observe in Fig. 1E that the curl filter is orthogonal to the divergence filter. For deformation two orthogonal masks are necessary for capturing distortion in all directions. For illustration in Fig. 1E, we show two deformation masks with maximal expansion axes aligned to 0 and $\pi/4$ degrees. To reduce corner artifacts, a circular envelope is applied on each of these masks.

Video Preprocessing

The ROIs were processed through a pruning method designed to discard noisy or background pixels; for example, pixels depicting cells or space beyond the cilia. This pruning method discarded pixels whose intensity changes fell below a set threshold. The threshold value was adaptive and specific to each ROI, as the intensities between ROIs varied greatly. For a single ROI, we computed the standard deviation σ_i of the time-varying intensity changes at each pixel and constructed a histogram of these standard deviations. We used the Kolmogorov-Smirnov distance metric to determine whether the histogram more closely resembled a gamma distribution or a Gaussian distribution. In the former case, we used the distribution's peak, or σ_{peak}, as the pruning threshold value, and discarded all pixels for which $\sigma_i < \sigma_{peak}$. If the distribution was better approximated by a Gaussian, we used the distribution's mean, or σ_{mean}, and discarded all pixels for which $\sigma_i < \sigma_{mean}$. We performed a connected component analysis on the remaining pixels and discarded all but those in the largest component. Differential invariants were computed and subsequent analysis was performed only on these remaining pixels.

Algorithm 1 Extract quantitative features of CM.

Input: List of CM video patches $\mathcal{X}$; number of AR dimensions q; number of frequency histogram bins κ.

Output: Quantitative representations of CM as autoregressive motion parameters and magnitude/frequency rotation histograms (deformation magnitude/frequency histograms omitted for brevity).

```
function CILIARY_MOTION_FEATURES($\mathcal{X}$, $q$, $\kappa$)
    $\mathcal{R} \leftarrow \emptyset, \mathcal{H} \leftarrow \emptyset, \mathcal{A} \leftarrow \emptyset$    ▷ Initialize empty sets.
    for $X \in \mathcal{X}$ do    ▷ For each video patch/ROI.
        $X = \texttt{preprocess}(X)$    ▷ Remove noisy pixels.
        $O_x, O_y = \texttt{optical\_flow}(X)$    ▷ $x$ and $y$ components of optical flow.
        $R = \texttt{rotation}(O_x, O_y)$    ▷ $R$ has same dimensions as $X$, $O_x$, and $O_y$.
        $\mathcal{R}.\texttt{append}(R)$
        $\mathcal{H}.\texttt{append}(\text{HISTOGRAMS}(R, \kappa))$
    end for
    $\mathcal{A} = \text{AUTOREGRESSIVE}(\mathcal{R}, q)$    ▷ AR parameters for all video patches are computed at once.
    return $[\mathcal{A}, \mathcal{H}]$
end function

function HISTOGRAMS($R, \kappa$)
    $R_m = \texttt{temporal\_gaussian\_smoothing}(R)$
    $H_m = \texttt{histogram}(R_m, bins = 100)$    ▷ Magnitude histograms.
    $R_f = \texttt{temporal\_fft}(R)$    ▷ Fast Fourier Transform.
    $H_f = \texttt{histogram}(R_f, bins = \kappa)$    ▷ Frequency histograms.
    return $[H_m, H_f]$
end function

function AUTOREGRESSIVE($\mathcal{R}, q$)
    $U, \Sigma, V^T = \texttt{svd}(\mathcal{R}, q)$    ▷ $\mathcal{R}$ has dimensions $(width * height) \times (\texttt{len}(\mathcal{X}) * frames)$. $\Sigma$ has dimensions $q \times q$. $V^T$ has dimensions $q \times (\texttt{len}(\mathcal{X}) * frames)$.
    $S = \Sigma V^T$
    $\mathcal{A} = \texttt{argmin}_{\hat{\mathcal{A}}} ||S_2^{(n)} - \hat{\mathcal{A}} S_1^{(n-1)}||_F$    ▷ $S_1^{(n-1)}$ is the first $n-1$ rows of $S$; $S_2^{(n)}$ is also $n-1$ rows of $S$, starting with the 2nd row.
    return $\mathcal{A}$    ▷ Autoregressive motion parameters.
end function
```

Algorithm 2 Use quantitative features to predict CM.

Input: List of autoregressive motion parameters $\mathcal{A}$; list of magnitude and frequency histograms $\mathcal{H}$; list of ground-truth patient labels (0 = healthy, 1 = abnormal) $\vec{y}$.

Output: CM predictions for each **video patch**.

```
function CLASSIFY_AUTOREGRESSIVE($\mathcal{A}, \vec{y}$)
    $\mathcal{Z} \leftarrow \emptyset$                          ▷ List of cross-validation classification accuracies.
    for $A_{train}, A_{test}, \vec{y}_{train}, \vec{y}_{test} \in$ cross_validation_sets($\mathcal{A}, \vec{y}$) do
        $SVM$.train($A_{train}, \vec{y}_{train}$)               ▷ Train Support Vector Machine (SVM).
        $accuracy = SVM$.test($A_{test}, \vec{y}_{test}$)
        $\mathcal{Z}$.append($accuracy$)
    end for
    return $\mathcal{Z}$.mean()
end function

function CLASSIFY_HISTOGRAMS($\mathcal{H}, \vec{y}$)
    $\mathcal{Z} \leftarrow \emptyset$                          ▷ List of cross-validation classification accuracies.
    for $H_{train}, H_{test}, \vec{y}_{train}, \vec{y}_{test} \in$ cross_validation_sets($\mathcal{H}, \vec{y}$) do
        $K_{train} =$ HISTOGRAM_KERNEL($H_{train}$)
        $K_{test} =$ HISTOGRAM_KERNEL($H_{test}$)
        $SVM$.train($K_{train}, \vec{y}_{train}$)               ▷ Train Support Vector Machine (SVM).
        $accuracy = SVM$.test($K_{test}, \vec{y}_{test}$)
        $\mathcal{Z}$.append($accuracy$)
    end for
    return $\mathcal{Z}$.mean()
end function

function HISTOGRAM_KERNEL($\mathcal{H}, w = 0.5$)      ▷ Here $w = 0.5$ here as we use only two histogram types. In the full paper, we use four histograms, thus $w = 0.25$.
    $\mathcal{D}_f \leftarrow \emptyset, \mathcal{D}_m \leftarrow \emptyset$
    for $i = 1$ to len($\mathcal{H}$) do
        for $j = i + 1$ to len($\mathcal{H}$) do
            $d_f = \chi^2(\mathcal{H}_f^{(i)}, \mathcal{H}_f^{(j)})$          ▷ $\chi^2$ distance for frequency histograms.
            $d_m = \chi^2(\mathcal{H}_m^{(i)}, \mathcal{H}_m^{(j)})$          ▷ $\chi^2$ distance for magnitude histograms.
            $\mathcal{D}_f$.append($d_f$)
            $\mathcal{D}_m$.append($d_m$)
        end for
    end for
    $\mu_f = \mathcal{D}_f$.mean()
    $\mu_m = \mathcal{D}_m$.mean()
    $\mathcal{K} =$ zeros(len($\mathcal{H}$), len($\mathcal{H}$))
    for $i = 1$ to len($\mathcal{H}$) do
        for $j = i + 1$ to len($\mathcal{H}$) do
            $\mathcal{K}_{i,j} = \mathcal{K}_{j,i} = \sum_{\alpha \in \{f,m\}} w \exp\{-\mu_\alpha \chi^2(\mathcal{H}_\alpha^{(i)}, \mathcal{H}_\alpha^{(j)})\}$   ▷ Create the kernel matrix $\mathcal{K}$ of pairwise weighted chi-square distances.
        end for
    end for
    return $\mathcal{K}$
end function
```

Algorithm 3 Perform majority vote on classification results to determine final classification accuracy.

Input: List of classifier predictions (0 = healthy, 1 = abnormal) of the ciliary motion in each patch $\vec{y}_{pred}$; associative array providing the indices of the patches derived from each patient $\mathcal{P}$.

Output: CM predictions for each **patient**.

```
function MAJORITY_VOTE($\vec{y}_{pred}, \mathcal{P}$)
    $\mathcal{W} \leftarrow \emptyset$                ▷ Dictionary mapping each patient $P$ to final CM prediction.
    for $P \in \mathcal{P}$ do
        $\vec{y}_P = \vec{y}_{pred}$[indices_of_patient($P$)]   ▷ $y_P$ contains CM predictions for ROIs from patient $P$.
        $\mu$ = sum($\vec{y}_P$)/len($\vec{y}_P$)                 ▷ $0 \leq \mu \leq 1$
        if $\mu > 0.5$ then
            $\mathcal{W}[P] = 1$                                   ▷ Abnormal prediction.
        else
            $\mathcal{W}[P] = 0$                                   ▷ Healthy prediction.
        end if
    end for
    return $\mathcal{W}$
end function
```

Supplementary Figures

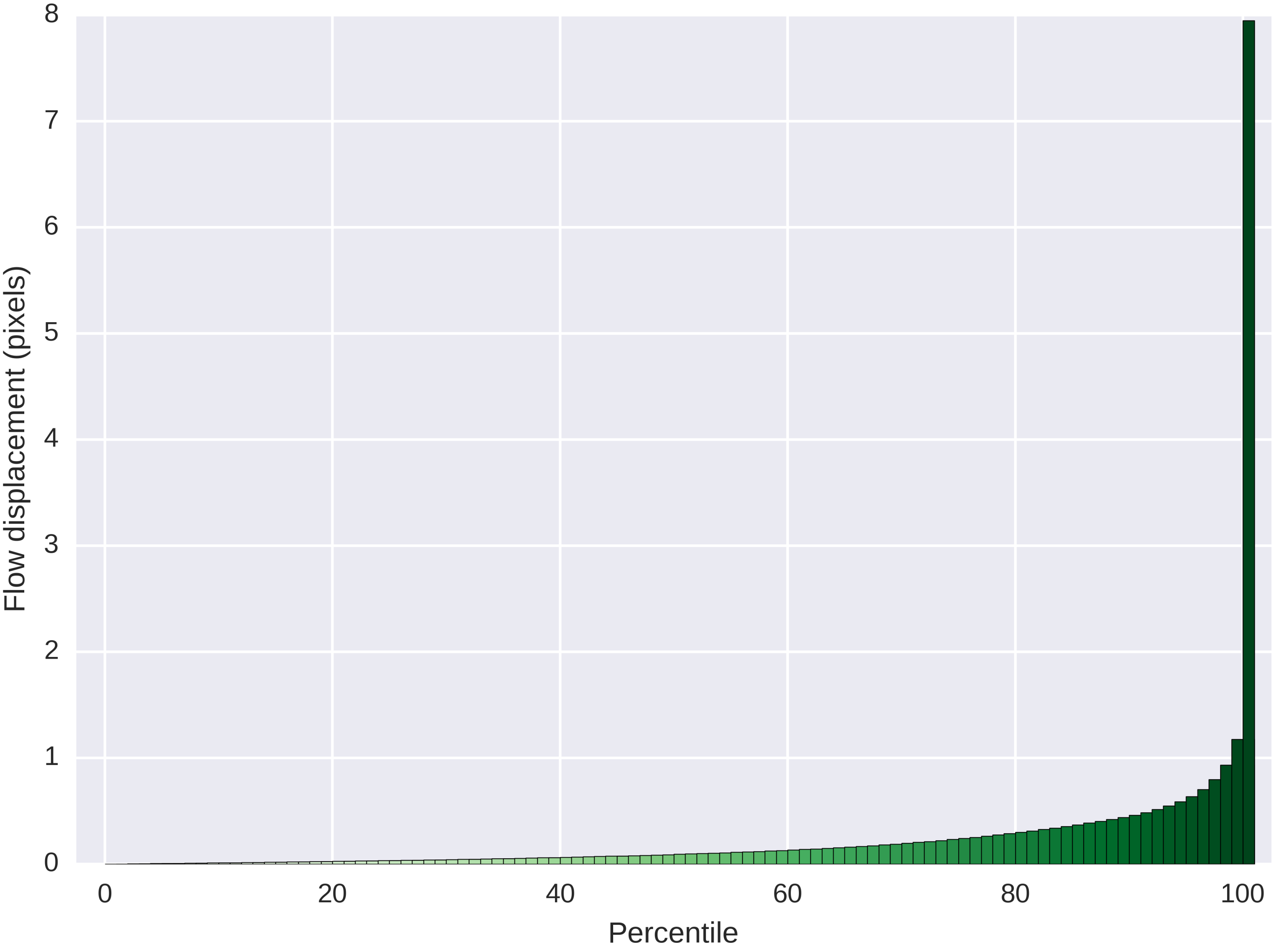


Figure S1. Aggregate optical flow displacement in CHP data cohort. These percentiles are aggregated across all optical flow computations made on the CHP cohort, indicating that fewer than 1% of optical flow displacements, or tracked motion between sequential frames, are greater than 2 pixels in magnitude.

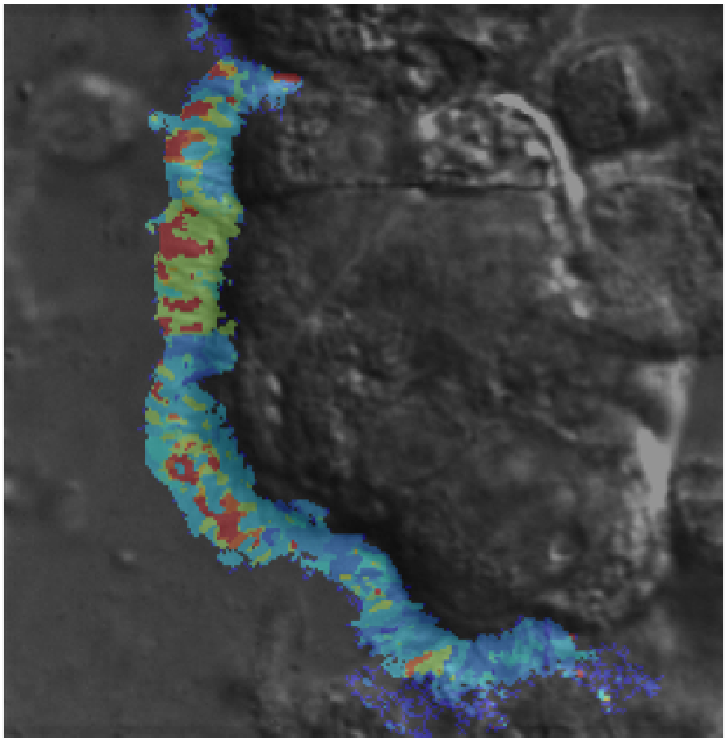


Figure S2. Pixel selection in an ROI. The heatmap overlay indicates the dominant frequencies at each pixel. Light blue indicates low-frequency motion (1-5Hz), where yellow and red indicate higher-frequency motion (5-15Hz). Pixels without color overlays were discarded by the adaptive pruning method.

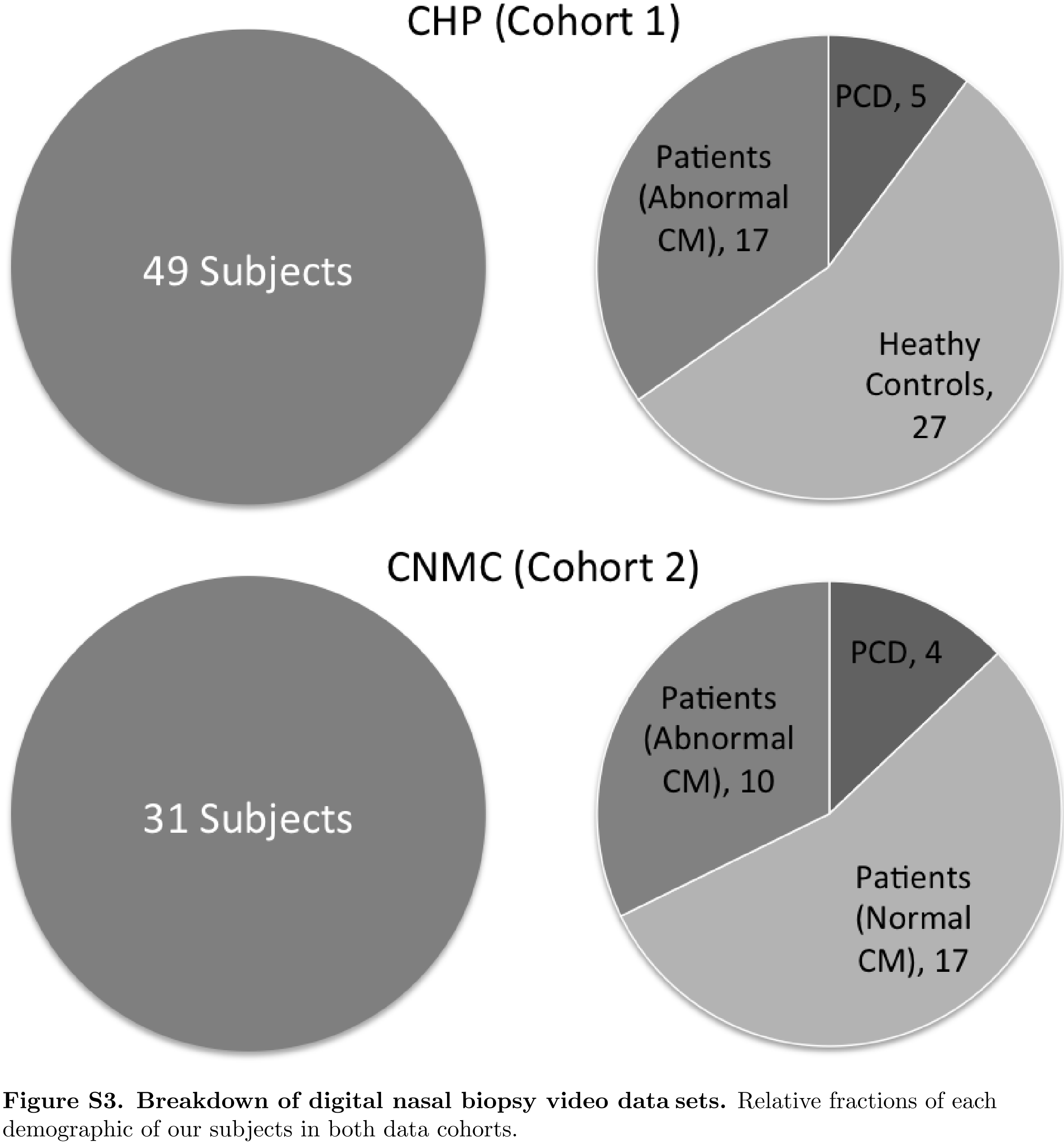


Figure S3. Breakdown of digital nasal biopsy video data sets. Relative fractions of each demographic of our subjects in both data cohorts.

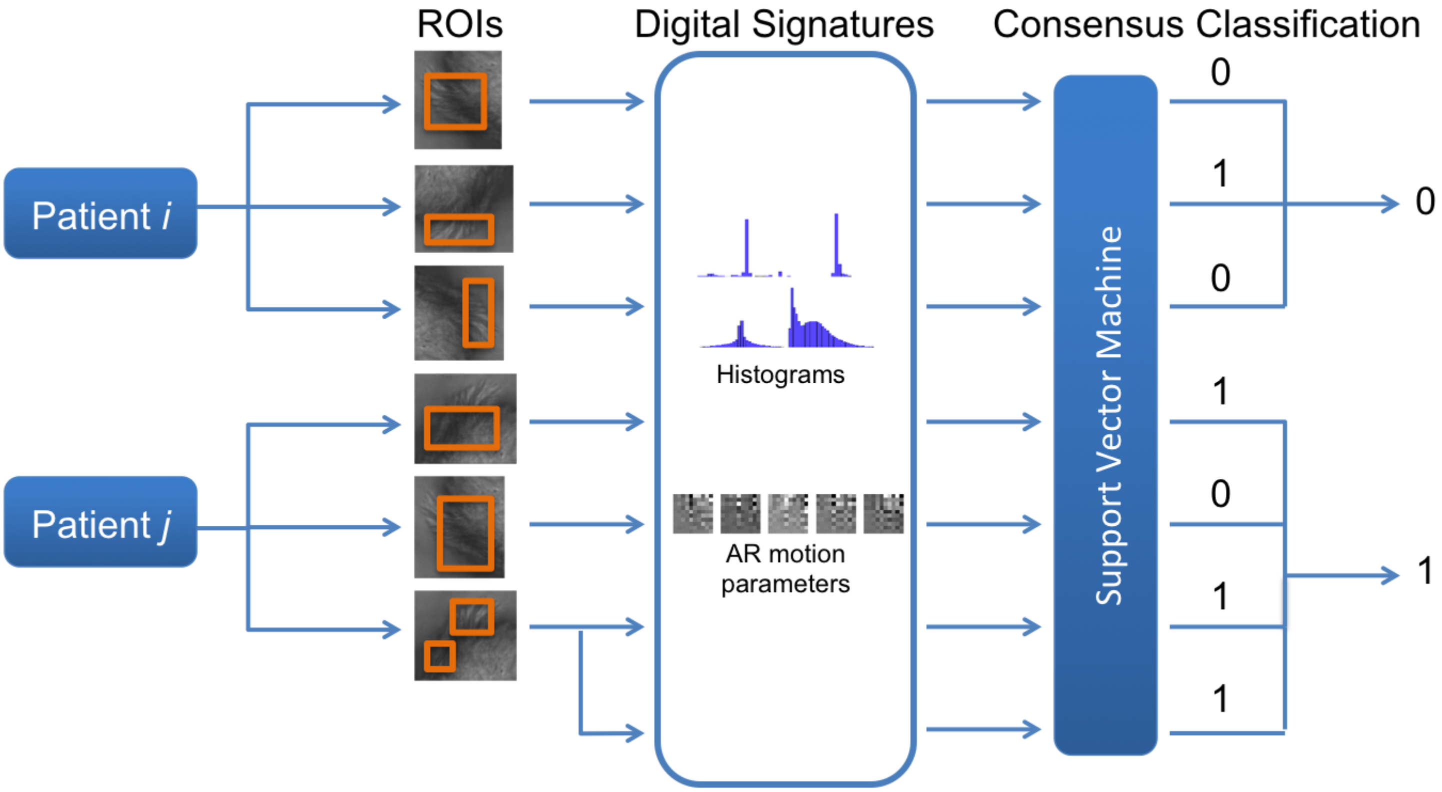


Figure S4. CM classification pipeline. Patient data, in the form of a handful of ROIs, is classified as normal (0) or abnormal (1) based on the method (histograms or AR). A majority vote, or "consensus classification," is performed using the ROI classifications for a single patient to predict the CM of the patient. All results are reported as predictions for each patient.

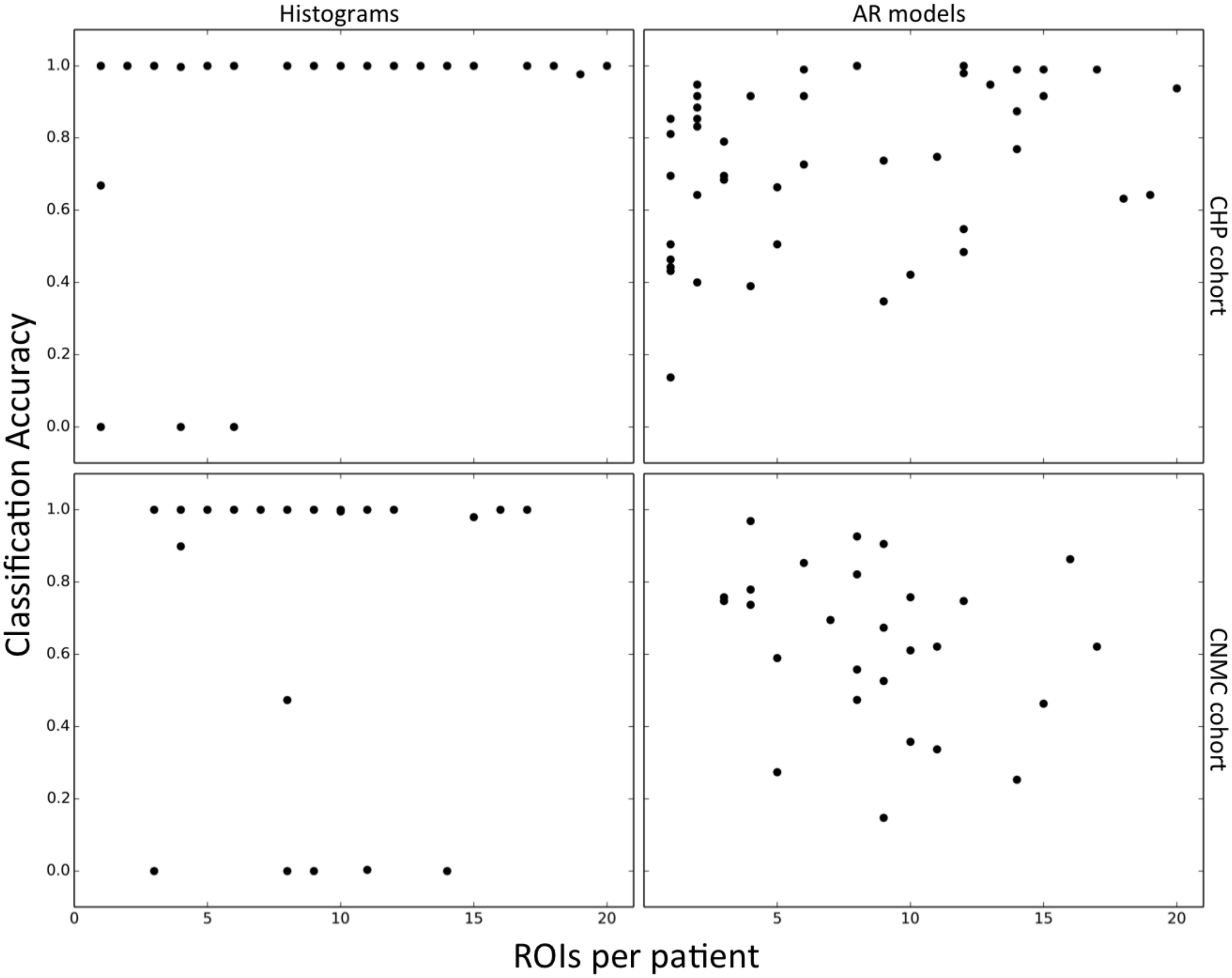
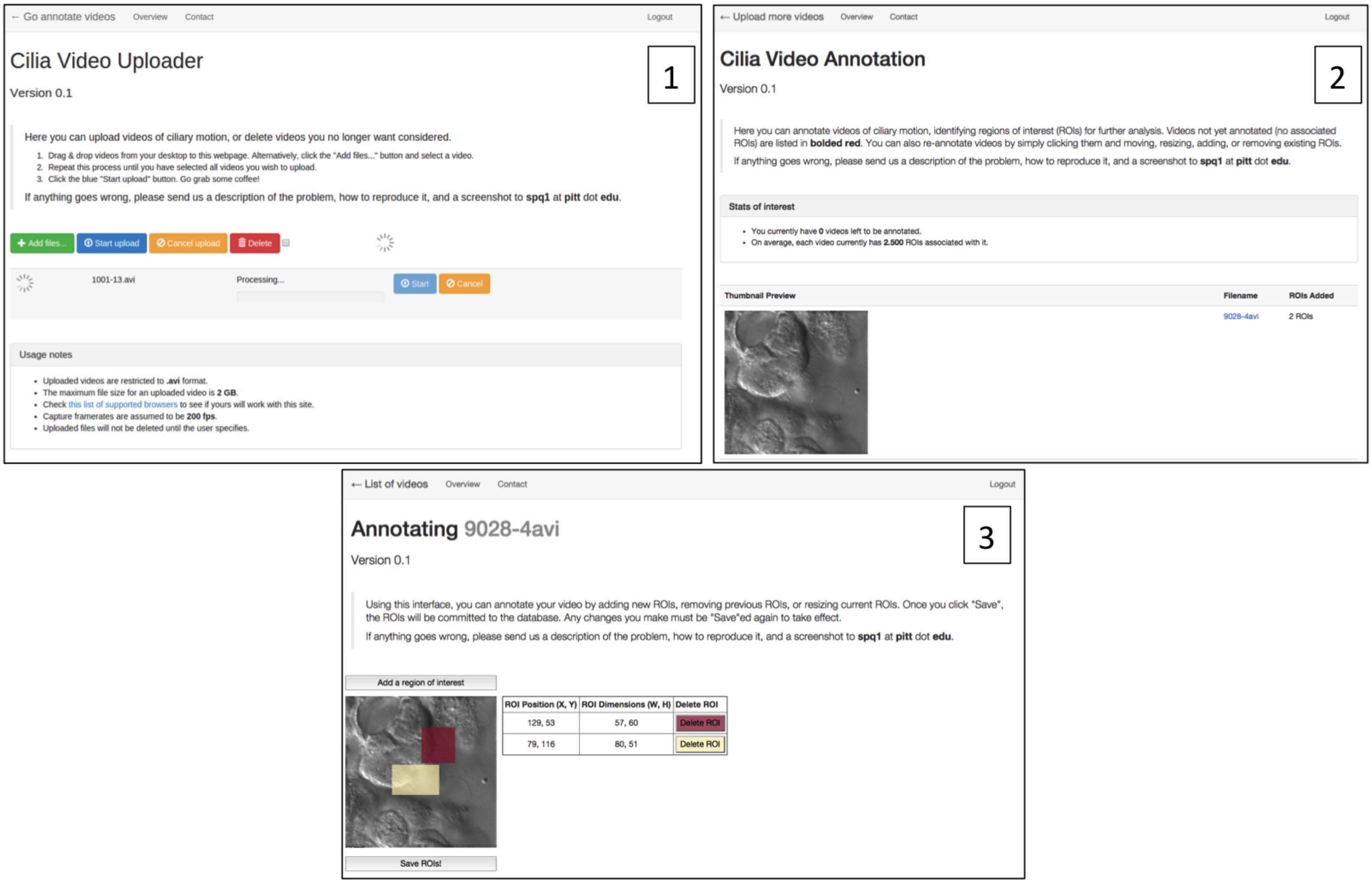


Figure S5. Classification confidence as a function of ROIs per patient. While the overall classification accuracy for each method is specified in Table 2, these plots show classification accuracy for individual patients as a function of the number of ROIs associated with that patient. 1.0 indicates the CM of that particular patient was always identified correctly (whether normal or abnormal); conversely, 0.0 indicates our framework consistently misclassified the CM of that patient. In the CHP cohort (top row), there is a clear trend toward perfect CM classification as the number of ROIs per patient increases; in the CNMC cohort (bottom row), this trend is much less apparent, even entirely absent, suggesting quality of ROIs is more important than quantity in classifying patient CM.



Cilia Video Uploader

Version 0.1

Here you can upload videos of ciliary motion, or delete videos you no longer want considered.

1. Drag & drop videos from your desktop to this webpage. Alternatively, click the "Add files..." button and select a video.
2. Repeat this process until you have selected all videos you wish to upload.
3. Click the blue "Start upload" button. Go grab some coffee!

If anything goes wrong, please send us a description of the problem, how to reproduce it, and a screenshot to **spq1** at **pitt** dot **edu**.

1001-13.avi Processing...

Usage notes

- Uploaded videos are restricted to **.avi** format.
- The maximum file size for an uploaded video is **2 GB**.
- Check this list of supported browsers to see if yours will work with this site.
- Capture framerates are assumed to be **200 fps**.
- Uploaded files will not be deleted until the user specifies.

Cilia Video Annotation

Version 0.1

Here you can annotate videos of ciliary motion, identifying regions of interest (ROIs) for further analysis. Videos not yet annotated (no associated ROIs) are listed in **bolded red**. You can also re-annotate videos by simply clicking them and moving, resizing, adding, or removing existing ROIs.

If anything goes wrong, please send us a description of the problem, how to reproduce it, and a screenshot to **spq1** at **pitt** dot **edu**.

Stats of interest

- You currently have **0** videos left to be annotated.
- On average, each video currently has **2.500** ROIs associated with it.

Thumbnail Preview	Filename	ROIs Added
	9028-4avi	2 ROIs

Annotating 9028-4avi

Version 0.1

Using this interface, you can annotate your video by adding new ROIs, removing previous ROIs, or resizing current ROIs. Once you click "Save", the ROIs will be committed to the database. Any changes you make must be "Save"ed again to take effect.

If anything goes wrong, please send us a description of the problem, how to reproduce it, and a screenshot to **spq1** at **pitt** dot **edu**.

ROI Position (X, Y)	ROI Dimensions (W, H)	Delete ROI
129, 53	57, 60	Delete ROI
79, 116	80, 51	Delete ROI

Figure S6. Web site proof-of-concept screenshots. We implemented a barebones proof-of-concept website for uploading, annotating, and analyzing videos of ciliary motion. This was used over the course of this study to enhance remote collaboration. This shows the three manual steps involved in uploading (1) and annotating (2 and 3) videos.

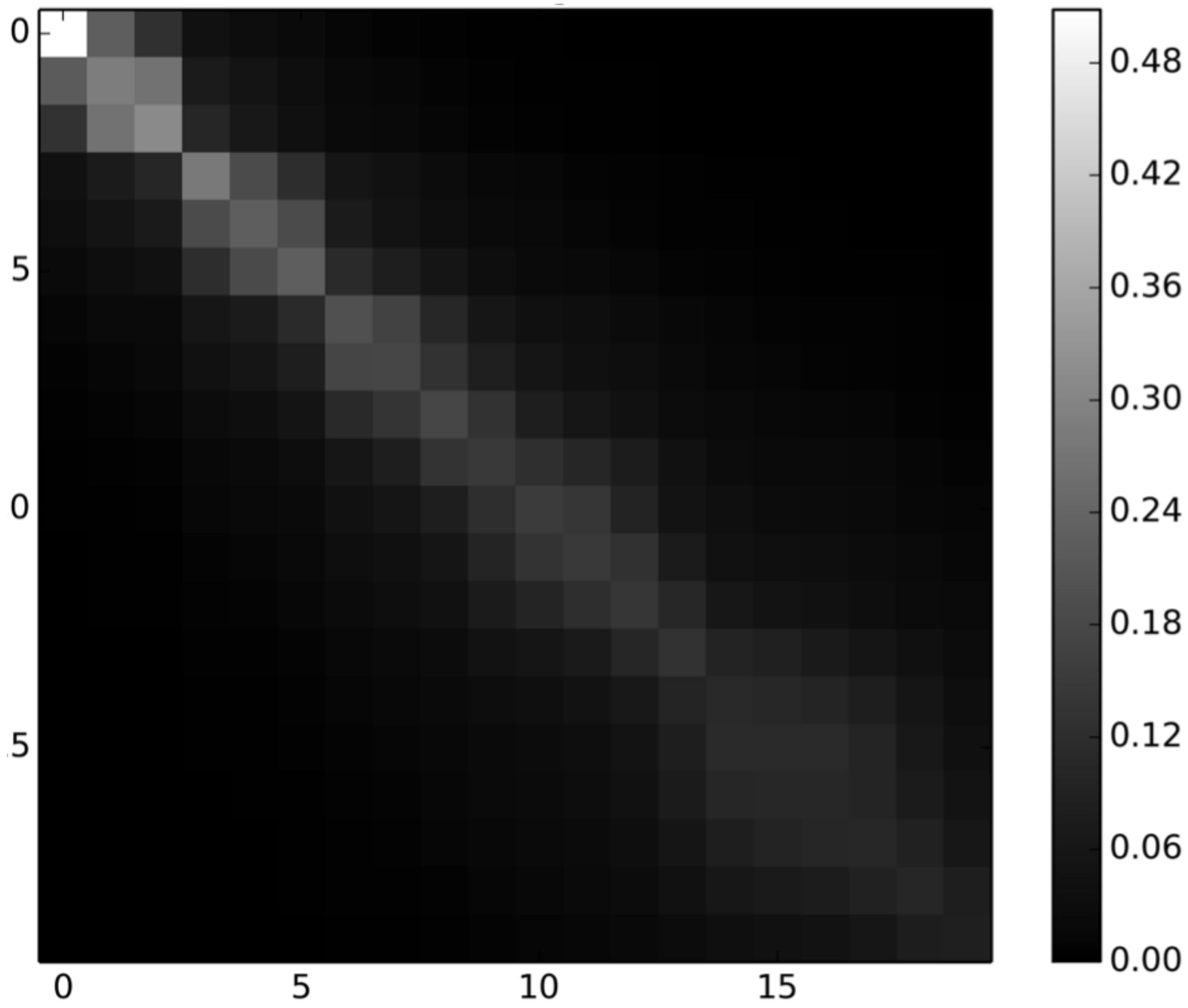


Figure S7. Pairwise angles between principal components of CM in AR models. Mean-squared average of all pairwise inner products of principal components derived from all image patches of ciliary motion used in this study. The large number of orthogonal (0 inner product) basis vectors provides strong evidence that all instances of CM occupy the same subspace.

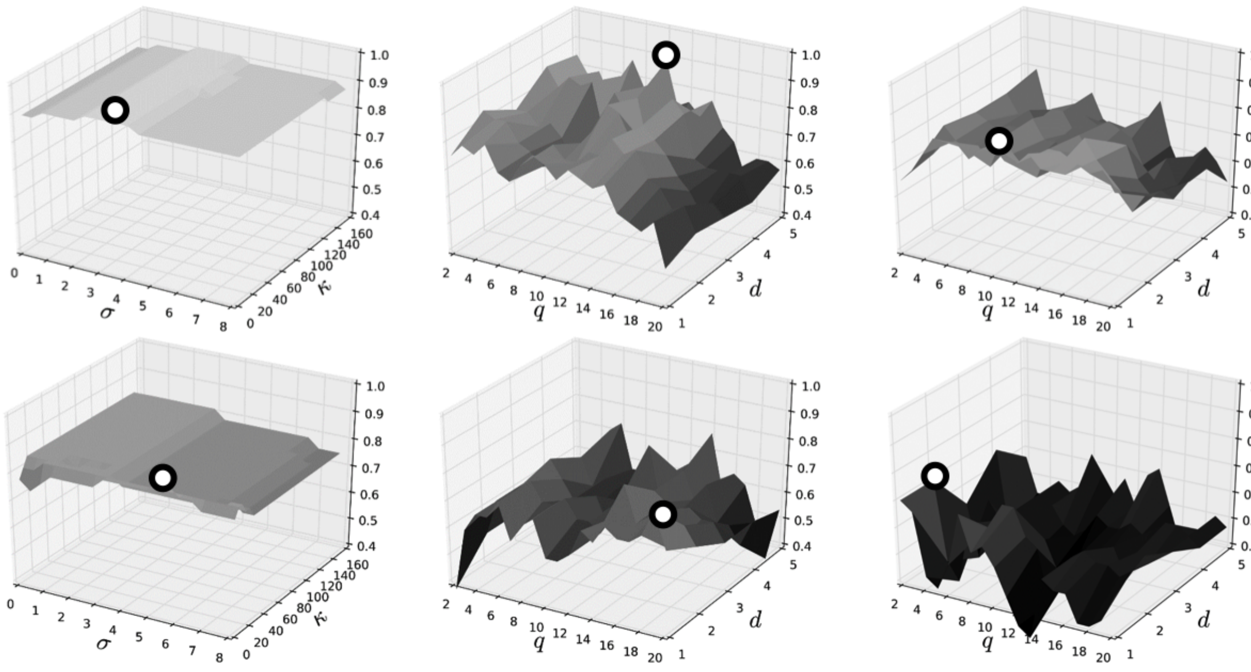


Figure S8. CM classification results of parameter scanning. Top row: scans over CHP cohort; Bottom row: scans over CNMC cohort. Each column indicates a different classification strategy: the leftmost is the histogram method; the middle is the AR model with rotation; the right is the AR model with deformation. Optimal parameter combinations are indicated by the white dot with a black border.

Supplementary Tables

Table S1. Constant parameters used throughout this study.

Parameter name	**Parameter value**	**Description**
Median filter	5	Used to smooth spurious frequencies in the heat map when building frequency histograms (RFH, DFH).
Maximum frequency	20 Hz	Maximum allowed frequency at each pixel when building frequency histograms; higher frequencies are suppressed.
Magnitude profile bins	100	Number of bins used in magnitude histograms (RMH, DMH).
ν	0.35	Used to train the SVM. See the `scikit-learn` documentation.
Cross-validation iterations	100	Number of independent and randomized cross-validations performed before averaging all results.
Cross-validation folds	10	Number of folds used in one iteration of cross-validation.
Maximum frames	250	Number of frames used from the original video data for building histograms and autoregressive parameters. Additional frames are ignored.
Autoregressive patch size	15	Width and height in pixels of the video patch used to train the autoregressive model.

Supplementary Movies

All videos are captured at 200 fps and formatted with a playback rate of 30 fps. All scale bars are 10 μm. Black boxes in each video depicts the ROIs selected for analysis.

Movie S1. Example of normal CM of nasal biopsy from control. These videos depict normal, healthy motion (Fig. 1A). The cilia beat in synchronized waves with a forward power stroke followed by a slower recovery stroke.

Movie S2. Example of abnormal CM of nasal biopsy from PCD patient. This video depicts the abnormal CM of a patient with PCD. Note the slow, dyskinetic and restricted CM and also immotile cilia.

Movie S3. Example of abnormal asynchronous and wavy CM. This video depicts wavy and asynchronous CM (Fig. 1A). Thus the cilia movement is not synchronized, and the beat pattern is symmetric with respect to the forward and backward strokes.

Movie S4. Example of abnormal CM with incomplete stroke. This video depicts wavy and asynchronous CM (Fig. 1A). Thus the cilia movement is not synchronized, and the beat pattern is symmetric with respect to the forward and backward strokes.

Movie S5. Example of abnormal CM with asynchronous beat and incomplete stroke. This video depicts dyskinetic CM (Fig. 1A) that is asynchronous and with incomplete stroke.

Movie S6. Example of a video capture artifact of extraneous tissue motion. This video depicts the effects of spurious sample motion. This motion is reflected in the optical flow computations, resulting in rotation and deformation quantities that capture both the depicted CM and the motion of the camera.

Movie S7. Example of a video capture artifact of poor camera focus. This video depicts the effects of blurry, unfocused video captures. This results in a very low signal-to-noise ratio, making the CM difficult to detect and quantify.

Movie S8. Example of a false-negative prediction. This video depicts ROIs of a airway sample with abnormal CM, but is consistently misclassified by our framework as having normal CM. In this case, there is a significant amount of spurious motion due to interference from nearby cells in both ROIs, causing the framework's misclassification.

Movie S9. Example of a false-positive prediction. This video depicts ROIs of normal CM our method consistently misclassified as having abnormal CM. Videos from a top-down perspective were rare in both datasets, but this perspective makes lateral motion much less apparent to our framework, resulting in misclassification.