# High Throughput Image Pipeline Documentation

By: Gunalan Natesan

The Mack Lab provided slides of of endothelial cells under variable laminar fluid flow, with 4 channels, each corresponding to DAPI stained nuclei, NOTCH1 proteins, tagged antibodies attached to Cavin proteins (indicative of caveolae positioning), as well as BAPTA-1-AM ester (indicative of calcium concentration), which is represented as the dye channel. I was tasked with processing the various channels by categorizing and correlating the various structures in each microscope image.

The projects that I have worked on are shown below. It is worth noting that this is *not* an exhaustive list of the tasks that I have undertaken.

**Project #1: Determining the orientation of the various nuclei on the DAPI channels of the slides with respect to the direction of variable laminar fluid flow (the horizontal axis) with a simple least squares approach.**

First, I preprocessed the relevant slides to create an unambiguous base layer of the shapes of the nuclei to run the preprocessing. As the DAPI slides were grayscale tiffs (Extracted from ZEN format), I converted the images to bitmaps such that all pixels above a variable threshold (32 out of 255 in this case) would be mapped to an “on value” (Ex: 255) while all other values would be converted to an “off value” (Ex: 0). This removes the slight blur that exists at the periphery of the DAPI stain to remove ambiguity about the nuclei borders.

My approach on this project was to first devise a way to count the number of nuclei that were on a particular slide, in a way such that one nucleus would be counted at a time.
My solution is reminiscent of Monte Carlo methods, as it takes a random point on the base nuclei layer as an argument and looks at its binary value. If the value is an “off value”, then the pixel selected is not part of a nucleus, and is thus ignored. If the value is an “on value”, then the pixel selected is part of a nucleus, and thus undergoes a flood fill over all of the “on values” from that point, which fills the area of the nucleus of the selected point. The nucleus counter ticks up by one, and all the values that have been covered with the flood fill have been tagged such that future calls over the tagged pixels no longer initiate flood fills or counts.This subroutine is looped for a number of iterations that corresponds to the area of the image, such that a single nucleus is categorized at a time.

The flood fill used to fully categorize the nucleus is a variant of the vertical span fill, such that it returns the pixel coordinates of the edges of the structure. The modified floodfill cannot track edges that form convex angles that have a tangent that is greater than 45 degrees from the horizontal axis of the image. However, given the nature of nuclei and its alignment to flow, there should be no such convex angles that exist on the horizontal on the nucleus [1]. This 1D fill is contained in the Vertical function and coordinate gathering is contained in the Vertcaps function. The Horizontal function calls the Vertical and Vertcaps functions along the horizontal component of the nucleus. These functions tag the selected nuclear structure and cover all of

the “on” values with a recognizable tag ((255,0,0) on a color .tiff layer generated from the initial structures.

The center of the nucleus (mean of x and y coordinates) are calculated and used to find the slope and constant parameters of a simple linear regression. The slope is calculated as the x values minus the x mean, multiplied by the y values minus the y mean all summed together, which is all divided by the sum of all of the x values minus the x mean. This parameter, represented by the variable b, can also be represented as the sample standard deviation of variables x and y divided by the sample standard deviation of x squared. The constant parameter is thus calculated by the slope multiplied by the x mean, all subtracted from the y mean. After calculated, a line constructed from the slope and constant would be overlaid over the center of the nucleus, and the length of the line segment of the line that lies on the nucleus would be calculated. In addition, another line is constructed from the same constant, but finding the negative inverse of the slope, which is also used to derive another line segment. The numerical lengths of each of these line segments are calculated, with the larger segment being designated as the major axis, while the smaller segment is denoted as the minor axis. These two perpendicular axes best represent the pseudo-ellipsoid shape of the nucleus, and are the numerical values that are used to determine the orientation of the nuclei on the slide. The longer of these two axes is laid over the corresponding nucleus in figure 1.3, where purple axes represent horizontal alignments and green axes represent vertical alignments. This processing is all encompassed in the lines function. It is worth noting that the lines function works remarkably well for the given nuclear shape, as it accommodates ellipsoids that can deviate from strict ellipses to a significant degree.

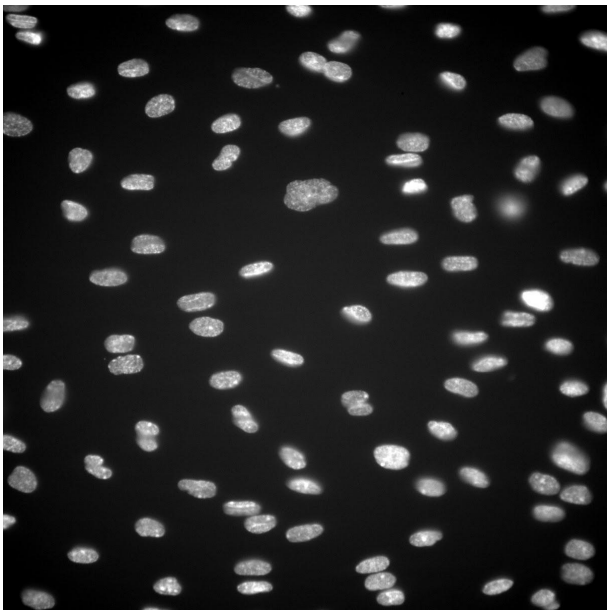

**Figure 1.1: Input is a .TIFF of a dense DAPI-stained nuclear layer of the slide**

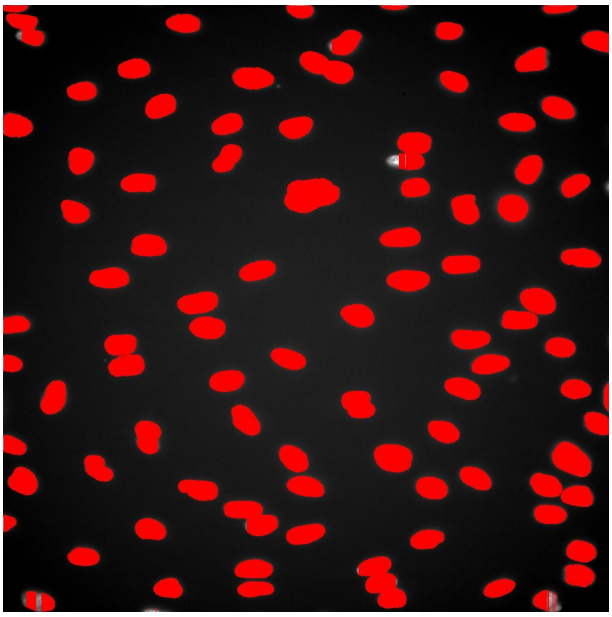

**Figure 1.2: First Output is a .TIFF highlighting nuclei of DAPI-stained layer**

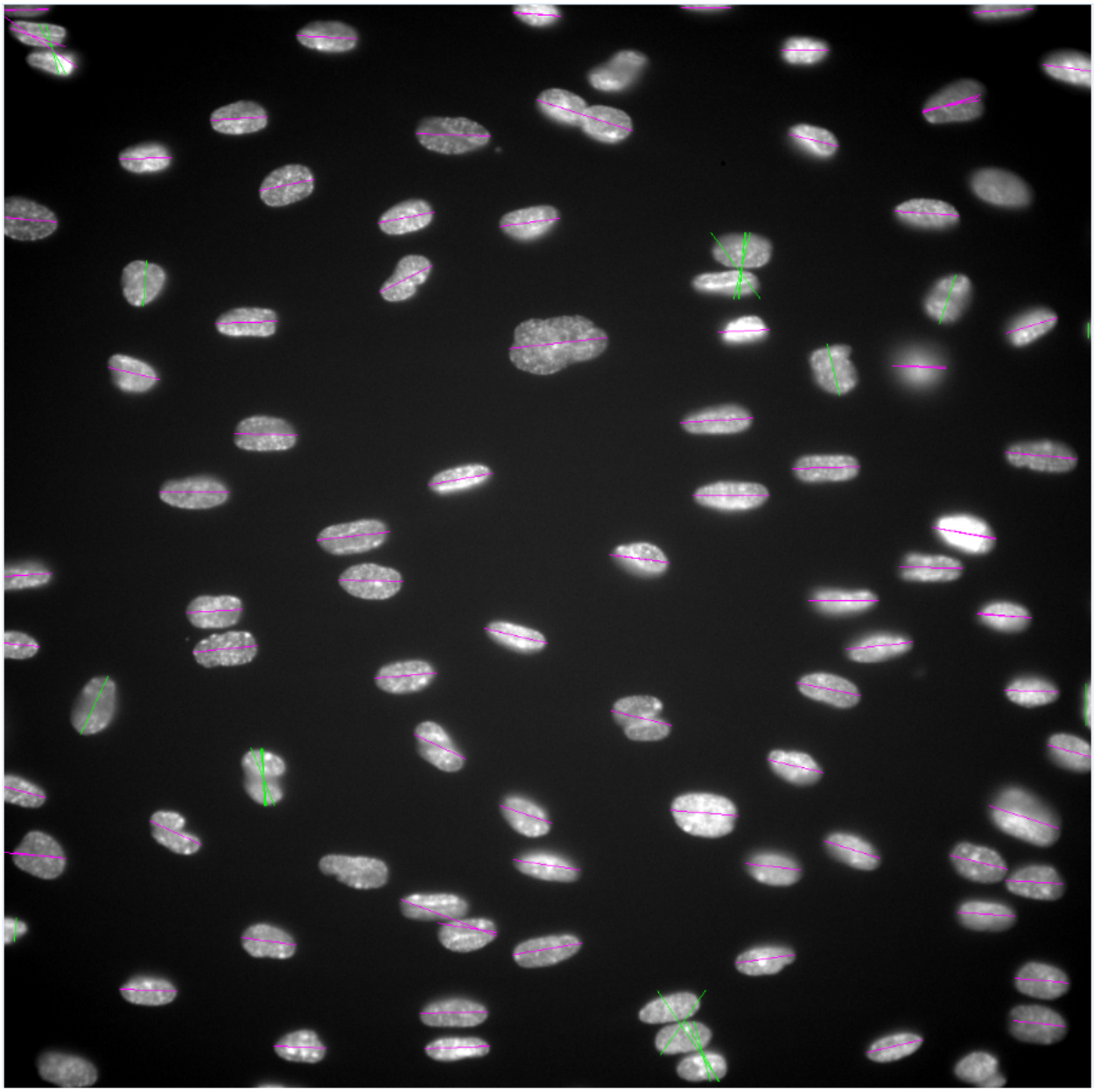

**Figure 1.3: Second Output is a .TIFF of major nuclear axes**

**Project #2: Segmenting endothelial cells on a given slide using the various channels that were given.**

It should be noted that the Cavin channels isolate and show caveolae specific proteins on the surfaces of cells. Thus, areas highlighted by the cavin channels tend to exclude areas that cell membranes run through on the full microscope image containing all of the constituent channels. In addition, the cell membranes of the endothelial cells were identified as various thin lines in the calcium dye channel, as opposed to the BAPTA-1-AM ester gradient that is present in the center of the cells in the dye channel. The approach I took was to isolate the cell membrane "signal" in the endothelial cells in the dye channel by removing "noise" such as the BAPTA-1-AM ester gradient. To do so, I used spatial distribution information that was present in the CAVIN channel to isolate the cell membrane.

First, I created two images based on the CAVIN and dye channels, where the CAVIN channel was represented as the gaussian blur of the tiff of the CAVIN channel with "on" and "off" values as described above (variable threshold was ~⅓ of pixel of maximum value). The dye image was a grayscale tiff of the dye channel for that particular image. The images were converted to matrices, and numpy commands were used to subtract the CAVIN matrix from the dye matrix to create a matrix that was equal to the dye matrix, except that it had removed any areas where CAVIN was found on the other slide. Since, as stated above, CAVIN presence tended to exclude the areas near the cell membrane, this produced a reasonable first approximation to find the membranes of the cells in the slide.

This matrix was then run through the cv2.equalize histogram function [2]. The function increases the contrast in an image by first creating a histogram of all of the pixel values in an image. It then finds cutoff points in the image where there are few to no pixels of a certain value (0 to lower bound and upper bound to 255) and removes these bounds from the image. The remaining nonzero part of the histogram is then stretched to cover the entire range of possible pixel values (0 to 255). These representative changes are also done in the image (moving each pixel in a direction away from the value of the mean in a manner proportional to the variance of the numerical pixel values of an image). This serves to magnify the contrast of the cell membranes much more than the remains of the dye, which, as a smooth gradient, do not pop out nearly as much. Then, further measures can be used to increase signal and reduce noise, which can include a low brightness filter and/or edge detection transforms and overlays.

The newly produced image has highlighted cell membrane structures, though these are riddled with discontinuities and interfering elements. These provide a rough outline of the various cell membranes, which can be used to reconstruct a complete structure by connecting them with other discontinuities. An approach that yielded promising results, and the one I picked, was to use convolutive morphological processing techniques [3]. These involve creating a base matrix, or kernel, of a certain size, with certain elements that make them up. Each element in the kernel is then multiplied by a corresponding element in a matrix sized region of the image. These are all summed up and assigned to a particular pixel in the post processed image. Of these techniques, I used the "opening" transformation (erosion followed by dilation) to remove interfering bright spots in the dye layer. This is then followed by a dilation with a kernel of variable dimension to connect the cell membranes in the various slides. The high flow slides have cells and membranes that follow the direction of flow more closely, and thus can be

connected with a high degree of accuracy if a horizontal kernel is used. The low flow slides, however, do not have an overwhelming directional influence, so a more rectangular kernel with high entropy elements must be used to segment the cells. Together, these various procedures create a serviceable template that can be further processed in the segmentation pipeline.

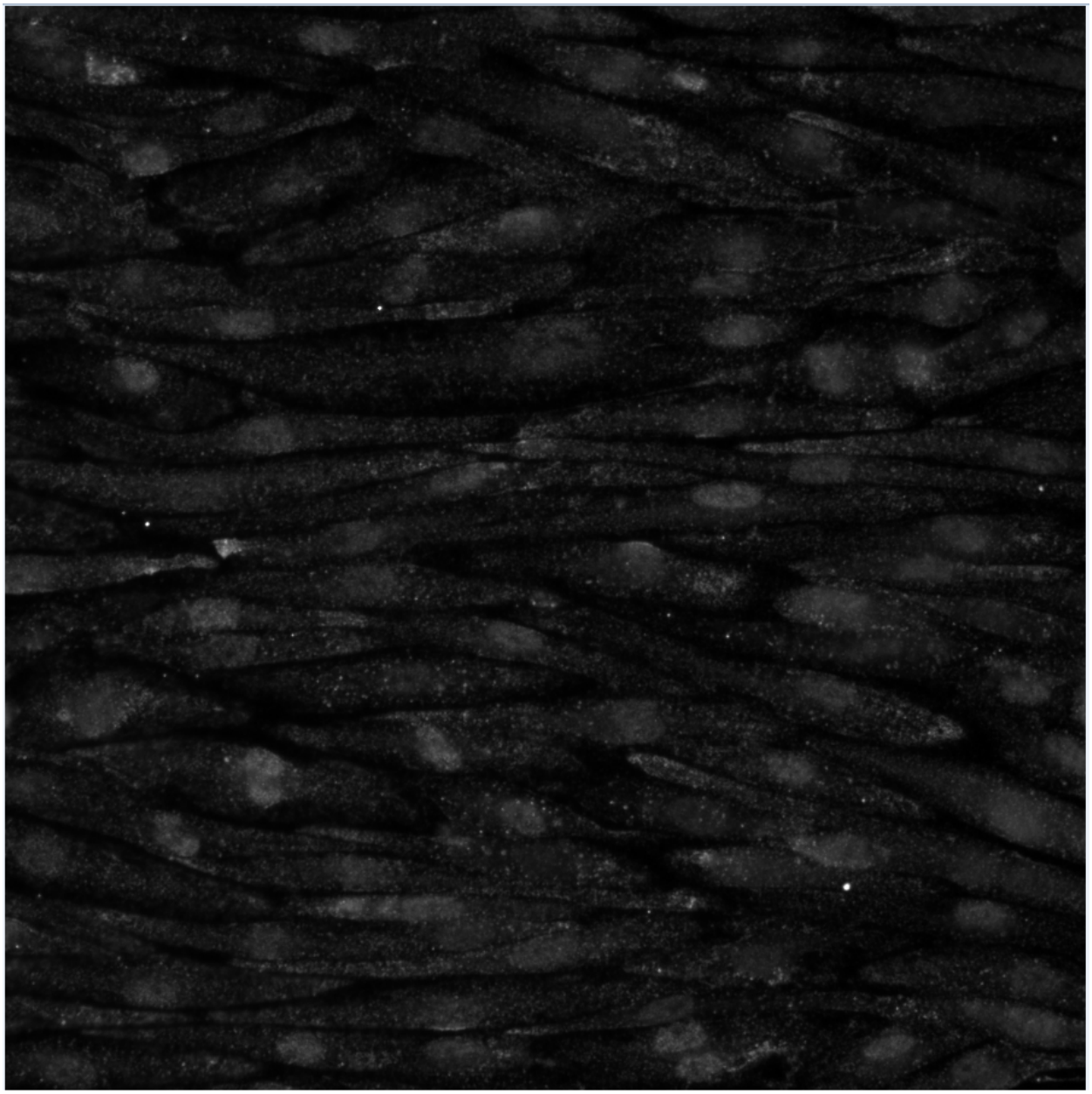

**Figure 2.1: First Input is a .TIFF of the CAVIN-tagged layer of the slide**

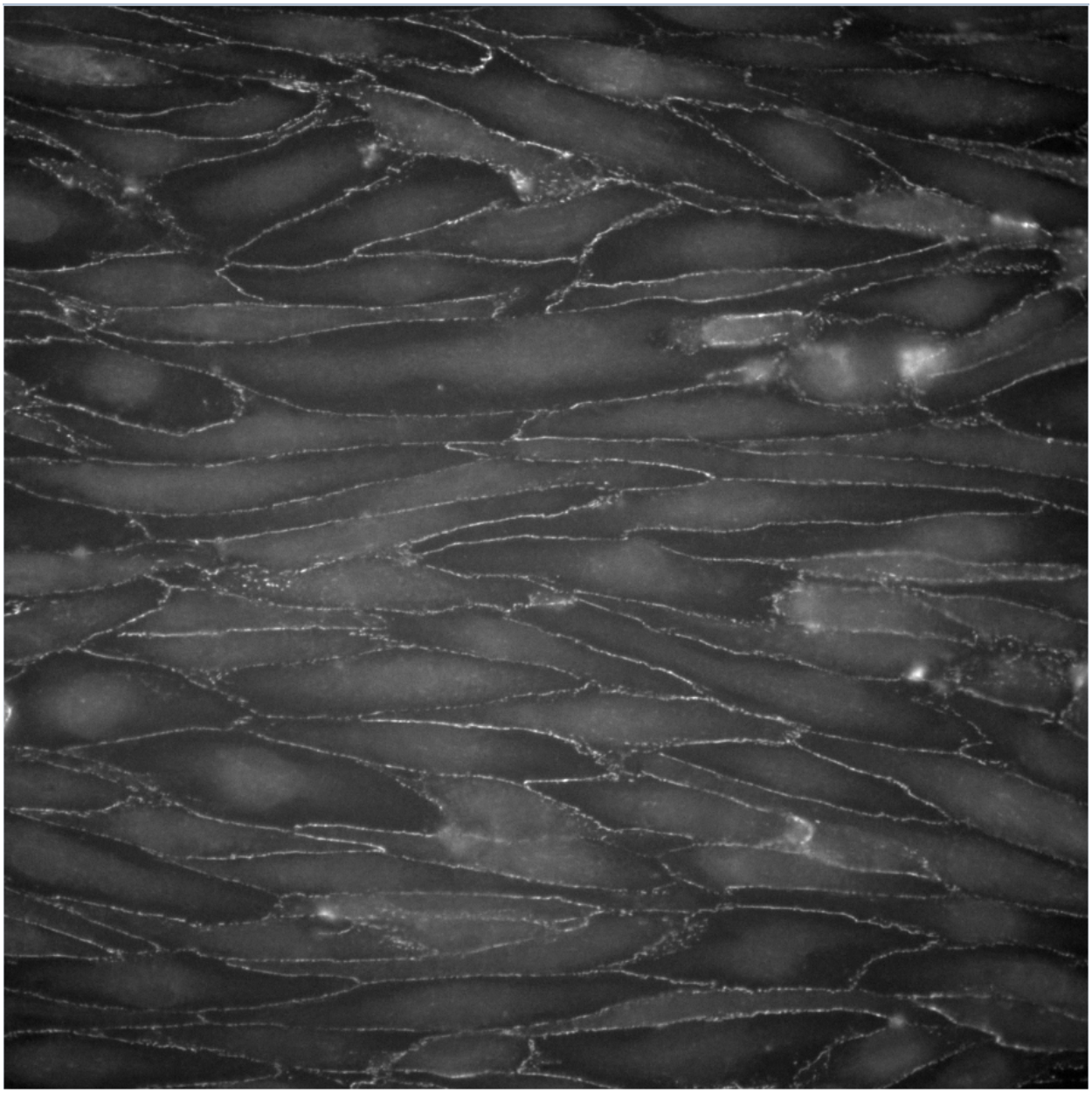

**Figure 2.2: Second Input is a .TIFF of the Calcium and Zo-1 dye layer of the slide**

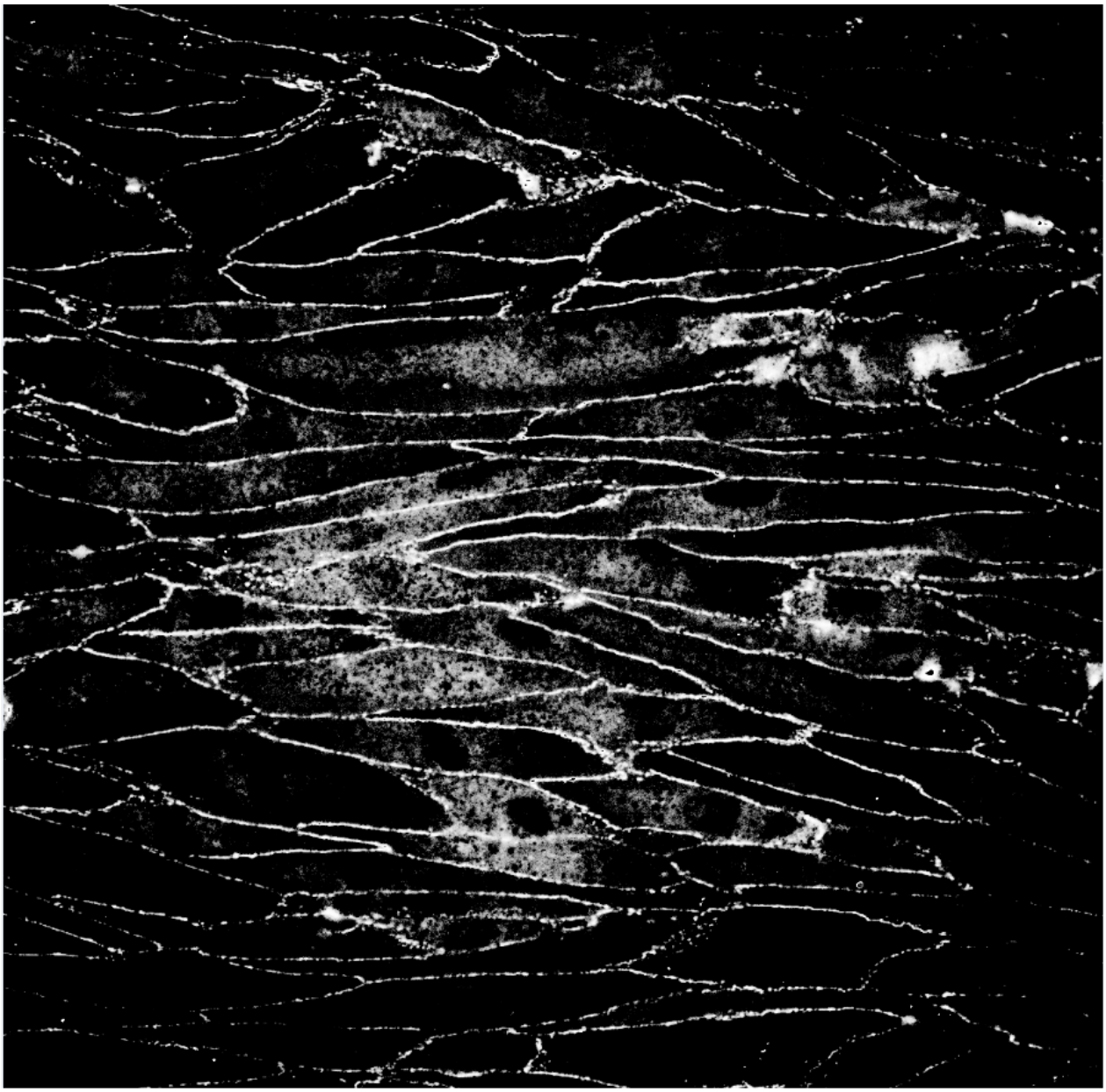

**Figure 2.3: Output is a .TIFF highlighting cell membranes on Calcium dye layer**

**Project #3: Streamlining the floodfill for use on the orientation quantification software.**

Various streamlining techniques could be used to drastically reduce the computational load of running the floodfill, at a slight expense for characterizing edge cases. However, since the nucleus is a topologically homogeneous structure throughout all of its occurrences in the slide, this does not effectively reduce the accuracy in gathering data. Since the first algorithm linearly traces the entirety of the inside of the nucleus, it loops through the height of the nucleus while looping through the length of the nucleus, which results in a worst case time of O(~pi*height*width) for an ellipse. The new variant that makes use of the fact that the nucleus is a smooth and continuous structure. That is, an edge between the inside and outside of the structure is more likely to be smooth without large jumps in continuity and sharp angles. Using this, I first used the existing vertcaps function to find the top and bottom of the nucleus for a certain x value. After this, I ran a loop to find the next edge value using the top and bottom edge values as reference instead of the center of the nucleus, while storing the new bottom and top values for each iteration. This measures the circumference instead of the area to reduce the time from O(~pi*height*width) to O(height + circumference) for an ellipse. Circumference can be approximated in the following function, where c ~= 2*pi*sqrt((height^2 + width^2)/2) where c is a pseudolinear function. The linear regression axial finder is then used to quantify the orientation.

.

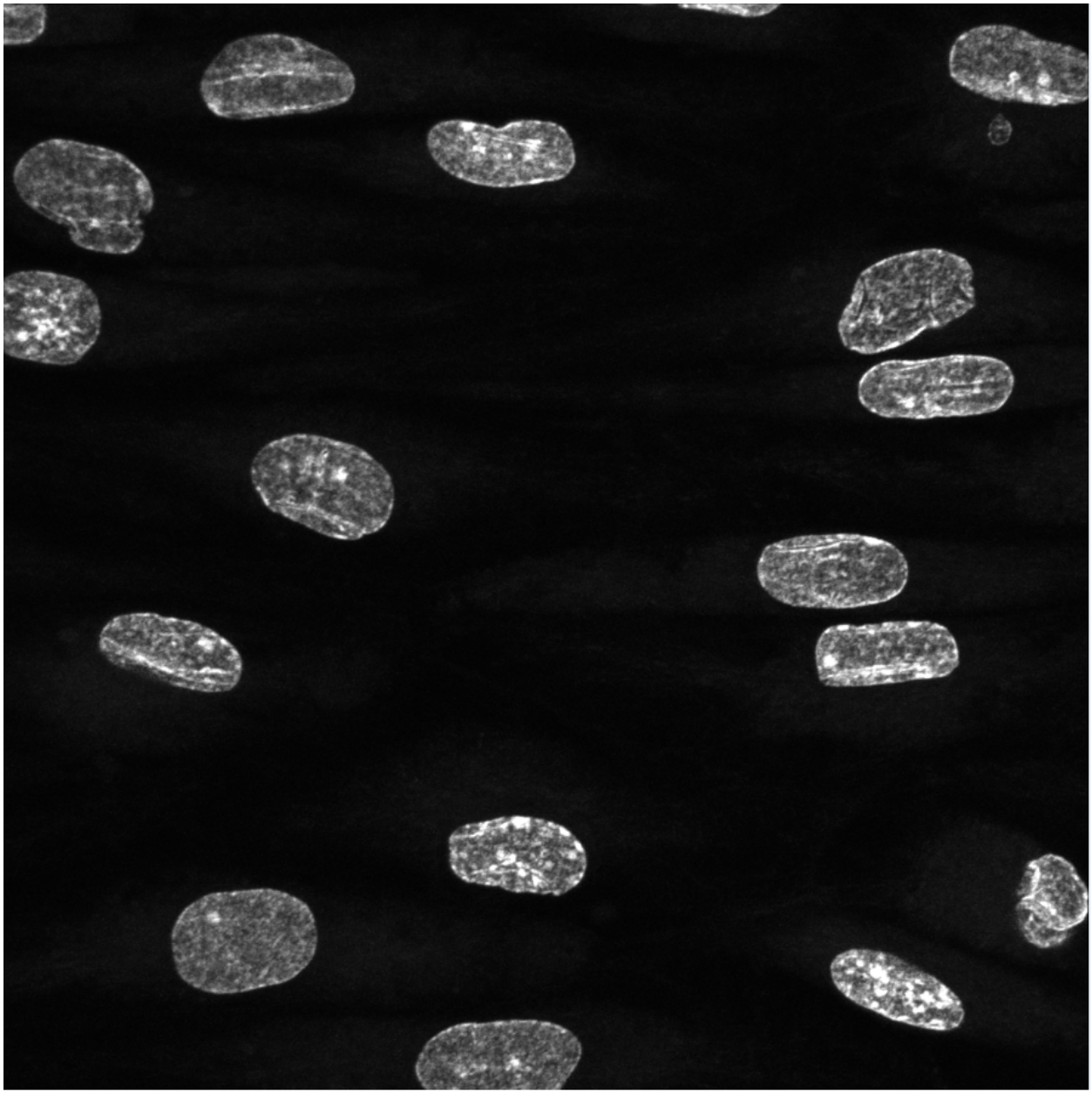

**Figure 3.1: Input is a .TIFF of a sparse DAPI-stained nuclear layer of the slide**

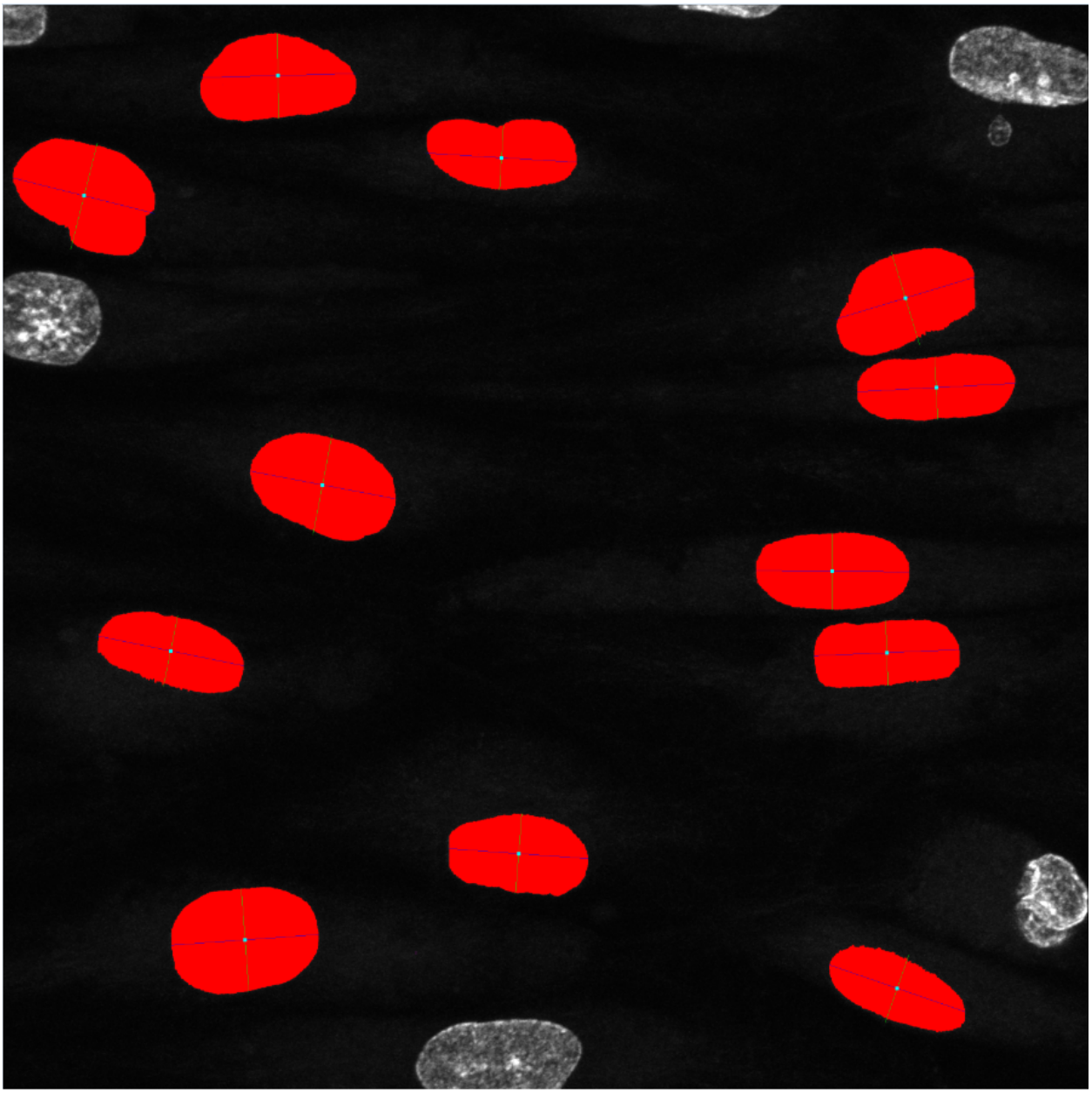

**Figure 3.2: First Output is a .TIFF highlighting nuclei of DAPI-stained layer**

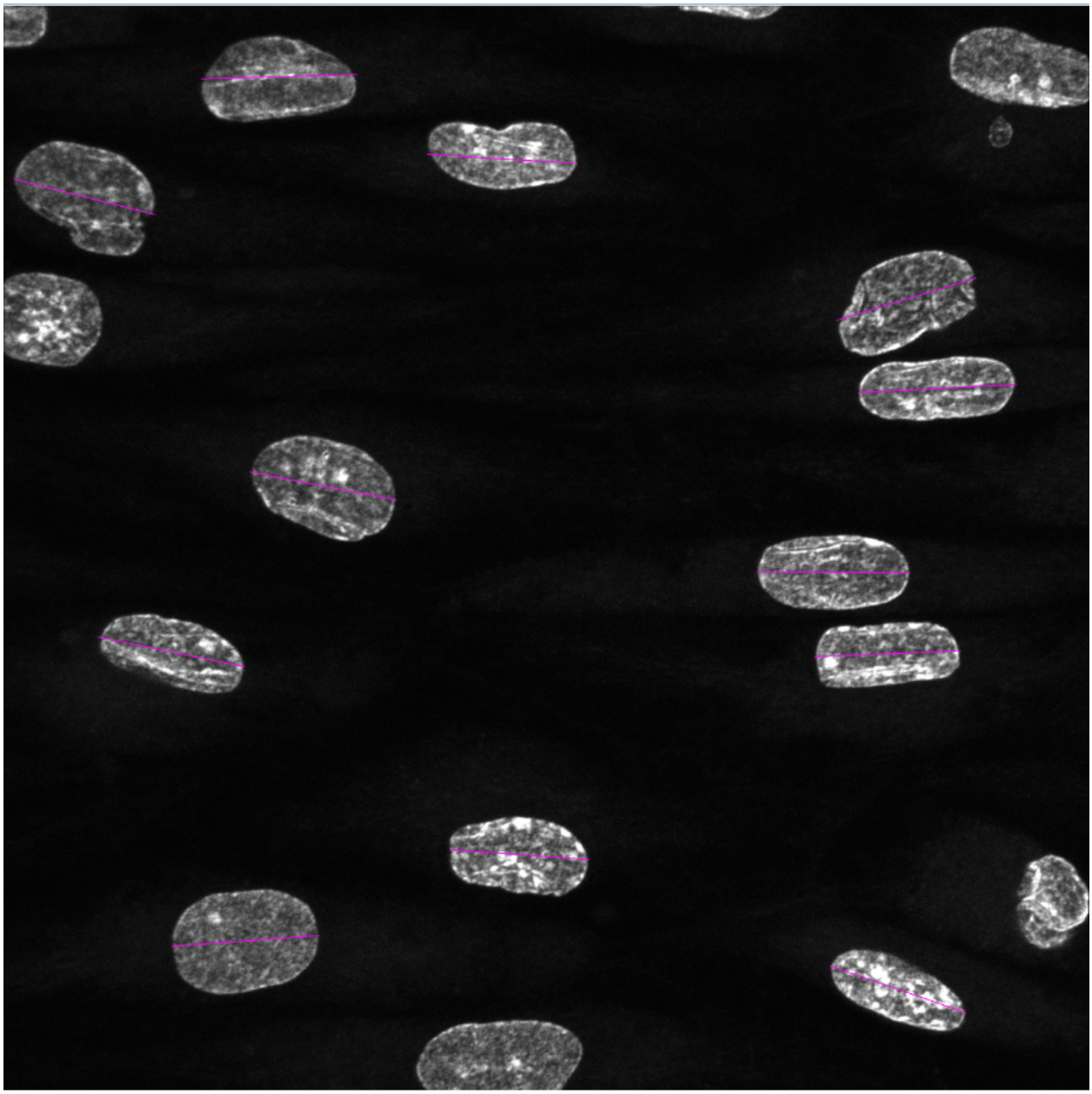

**Figure 3.3: Second Output is a .TIFF of major nuclear axes**

**Project #4: Determining the orientation of the various nuclei on the DAPI channels of the slides with respect to the direction of variable laminar fluid flow (the horizontal axis) with PCA.**

The alteration in this line of code is to replace the method of linear interpolation with from simple linear interpolation to Principal Component Analysis. Thus, after the various edge data points are collected, a new function is used to compute the principal components. These principal components consist of eigenvectors which specify the net directions of alignments of the various points for each dimension. These eigenvectors form an orthogonal basis, in which each basis/eigenvector has an eigenvalue that corresponds to the degree of alignment of the points along that given vector. Thus, the eigenvector with the largest eigenvalue has the greatest alignment with the given data, the eigenvector with the second largest eigenvalue is perpendicular to the already established vectors, and has the second greatest alignment, and so on.

The PCA is calculated by arranging the data points into a matrix and computing the mean for each dimension. The data matrix is then centered by subtracting the mean for each dimension from each of the appropriate points. The covariance matrix is thus calculated with this data, and taking the eigenpairs of the covariance matrix yields the principal components and their relative magnitudes. The eigenvalue with the largest eigenvalue is the vector of greatest alignment, which is in this case equivalent to the major axis of the points on the ellipse gathered. The vector is thus converted from a normalized (x,y) coordinate into a slope to be stored. The entire PCA procedure takes place in the PCA function that I have defined.

This is, mathematically speaking, a more robust procedure than simple linear regression. However, deviations in the shape of the nucleus make the linear regression more accurate due to the fact that many of the nuclei consist of multiple ellipsoid shapes that have fused together. This throws off PCA, which expects inputs to be directly linearly correlated. However, the more simple regression weights the points nearer to centers of mass of the nuclei, which allows for more accurate nuclear categorization. Therefore, linear regressions tend to be a more useful approach in nuclear orientation quantification than PCA when using pseudo-ellipsoid circumference points.

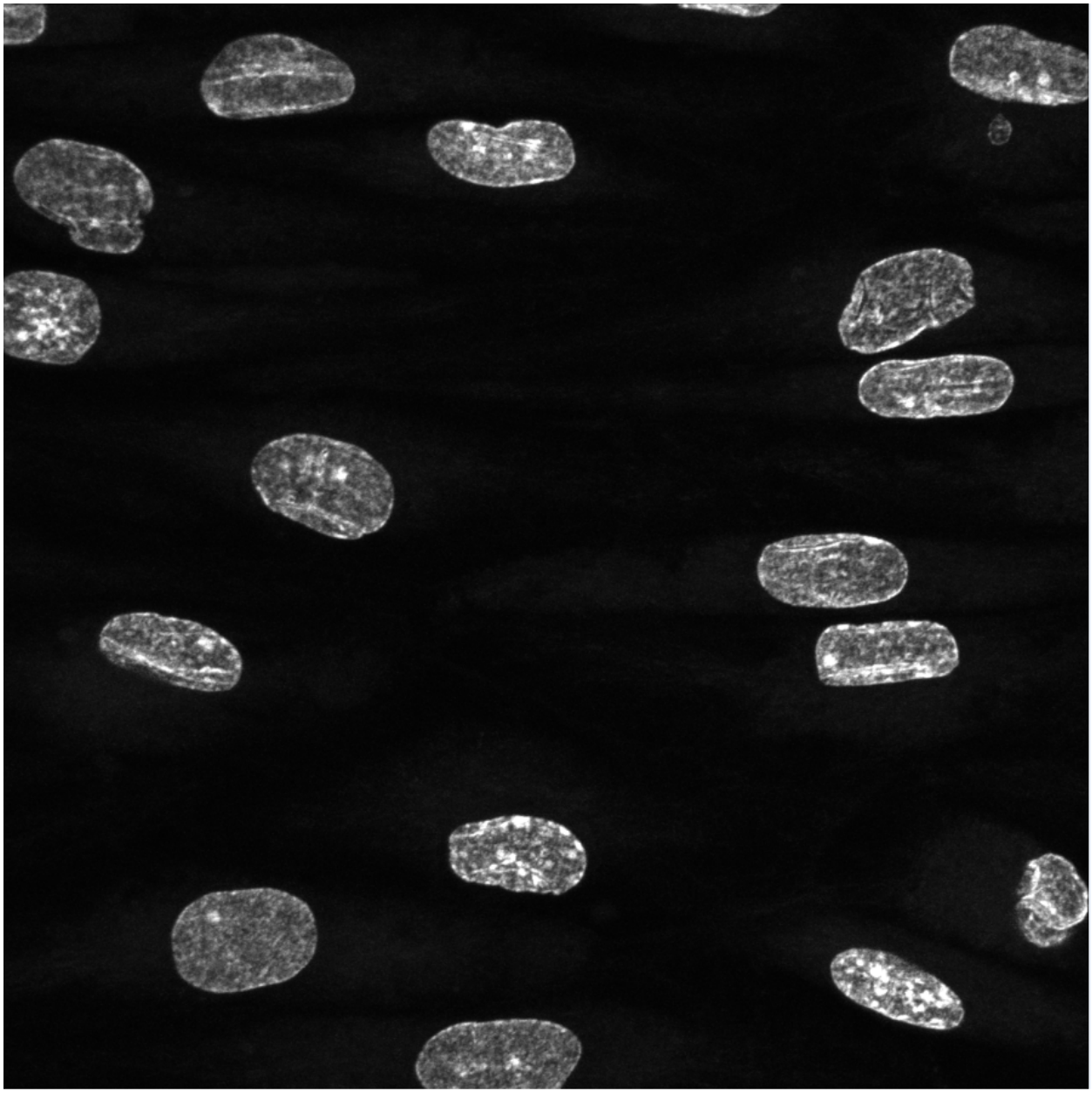

**Figure 4.1: Input is a .TIFF of a sparse DAPI-stained nuclear layer of the slide**

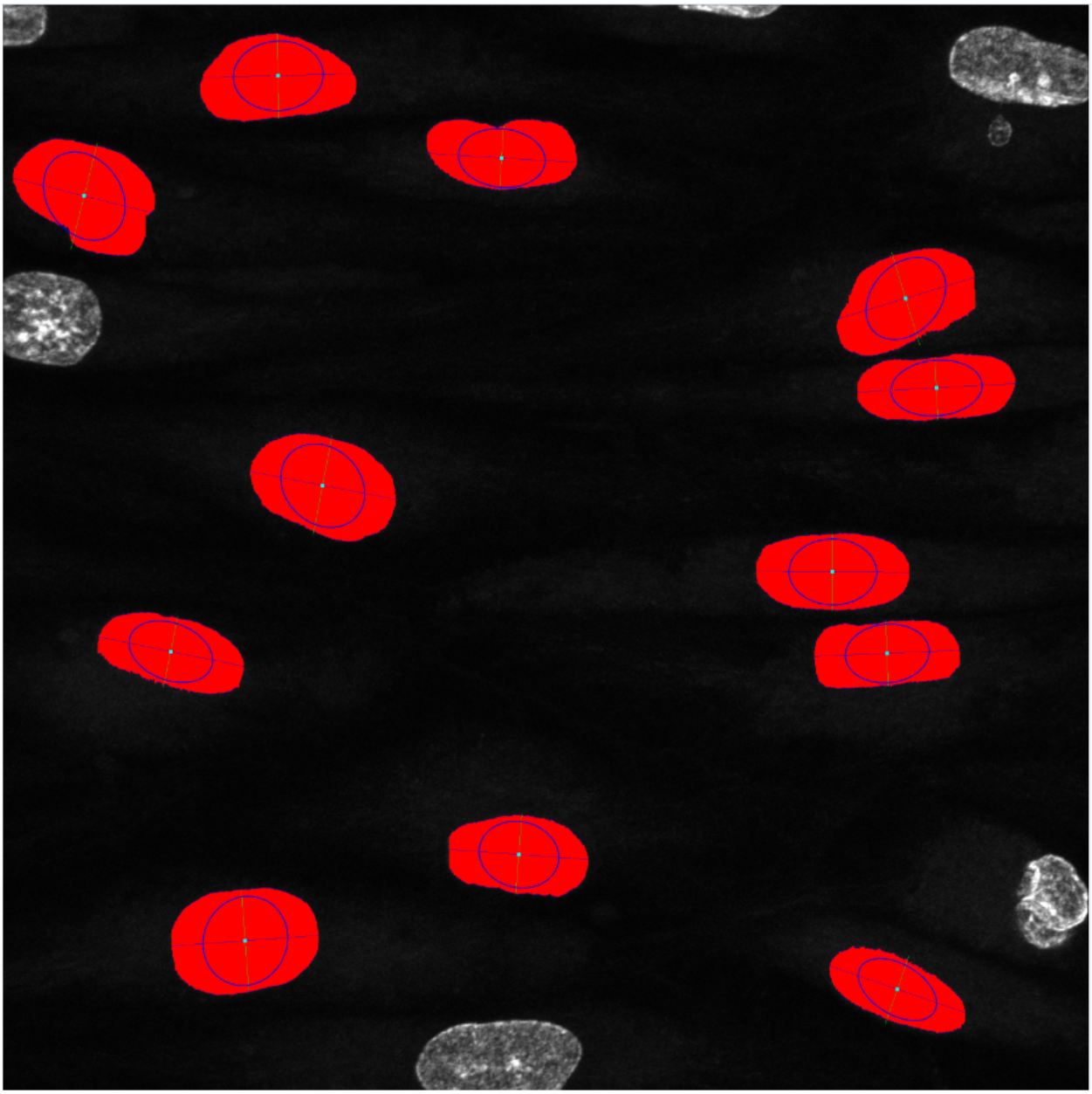

**Figure 4.2: Output is a .TIFF with highlighted nuclei, axes, and a PCA result overlay**

**Project #5: Reconstructing and Homogenizing Internal Golgi apparatus structure.**

The Mack lab began imaging Golgi structures of the Aortic Vascular Endothelial Cells that are used to measure response to laminar fluid flow. As such, images of the Golgi structures were captured on separate Golgi channels. These structures, unlike the nuclei that were measured before, were represented on the channel as a discontinuous collection of discrete structures and points. Finding the orientation of these discontinuous and discrete Golgi points would not be possible with the same methods used to characterize the smooth and continuous nuclei on the DAPI channel.

As such, some preprocessing was required to isolate the regions of interest where the central Golgi apparatus was located. The activated dye that was used to tag the Golgi apparatus was distributed throughout the cell though it centered on the central Golgi apparatus. This channel was then converted to a grayscale tiff that was run through the histogram equalization function, which highlighted contrasting areas throughout the slide. This equalization took pixel values that corresponded to areas of the slide that were saturated with dye, and increased those to a saturated level within the image as well. The slide then underwent a pixel value truncation, where all values that were smaller than a variable cutoff (~127?) were converted to zero, leaving behind only regions of saturated dye.

This was then processed with a morphological dilation, to help homogenize and connect the bright spots that represented the central Golgi apparatus. A standard floodfill is then used to remove holes in the internal structure of each of the processed Golgi apparatus images. A morphological erosion is then applied to normalize the size of the structures in the image. This results in a structure that overlaps the region of the central Golgi, but is slightly larger (dilated by a small degree in all directions, resulting in a statistically insignificant error), and visually exhibits fewer internal perforations. This structure can be quantified more easily with fill methods, since it can be more easily treated as a continuous and homogeneous structure. The linear regression method of evaluation produces a consistent and useful result when used on the constructed Golgi structure.

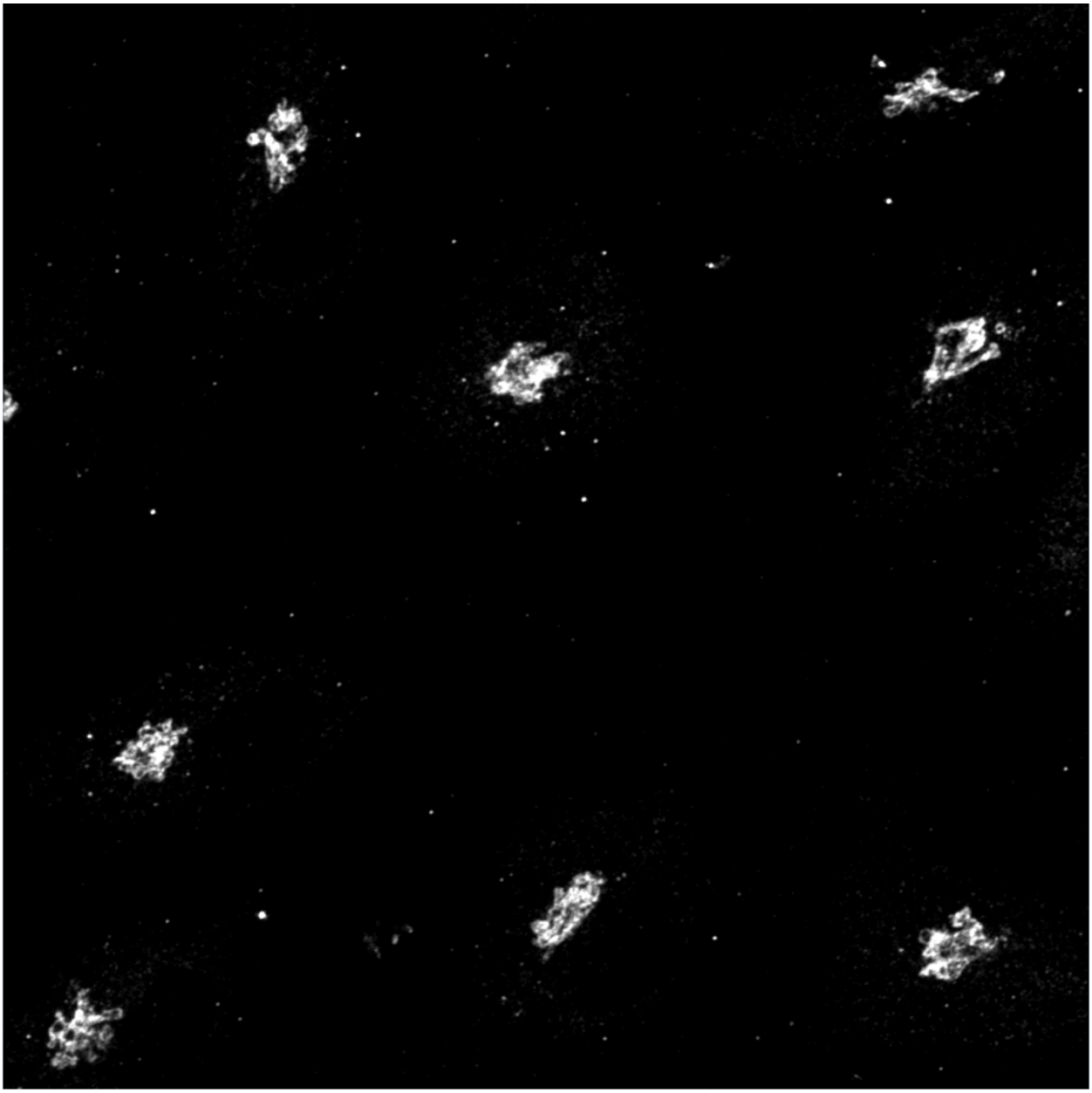

**Figure 5.1: Input is a .TIFF of the Golgi Apparatus Layer of the Slide**

**Figure 5.2: First Output is a Normalized .TIFF of the Golgi Apparatus Layer of the Slide**

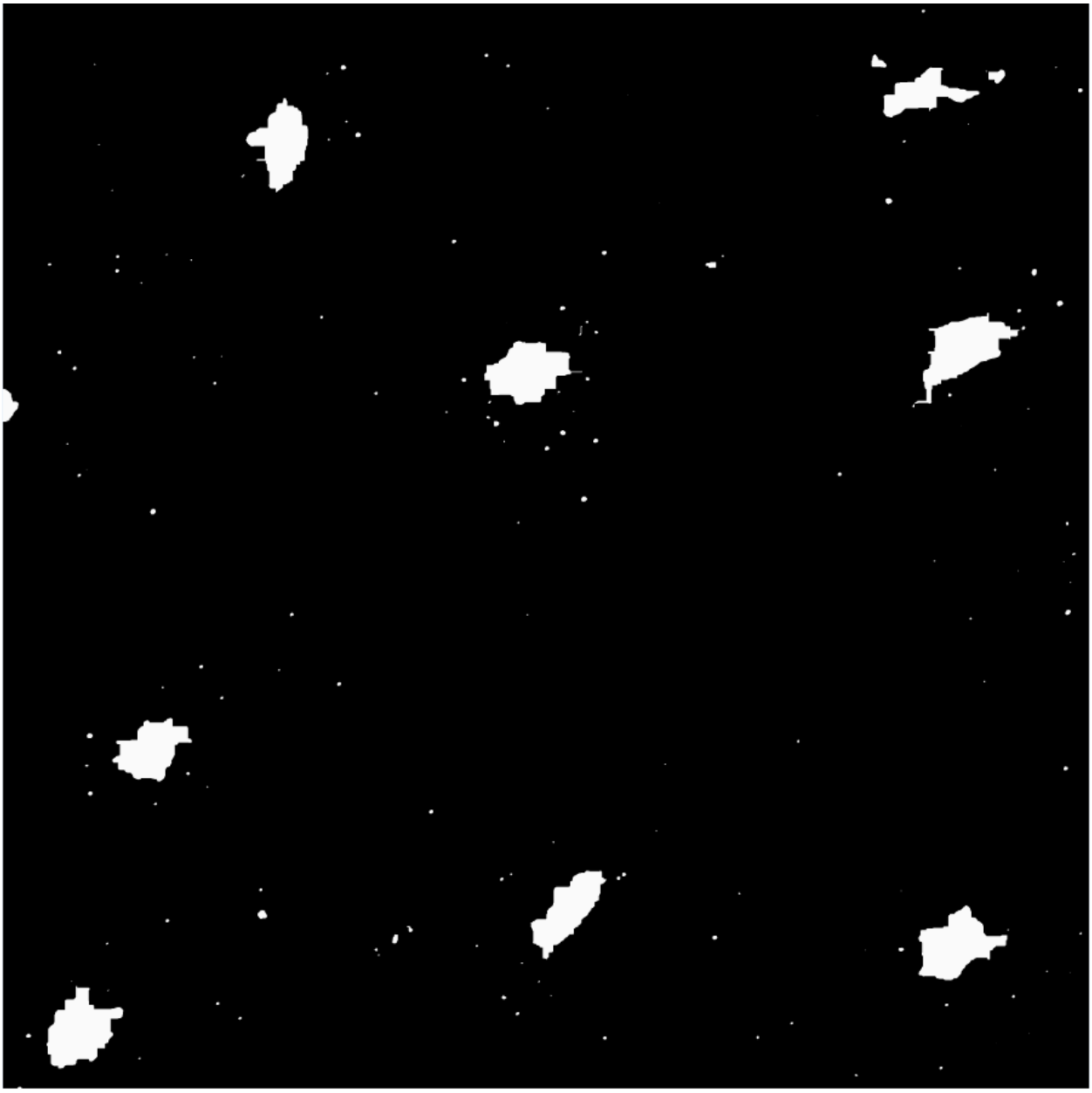

**Figure 5.3:Second Output is a Homogenized Normalized .TIFF of the Golgi Layer**

**Project #6: Categorizing degree of overlap between Golgi Apparatus and Nucleus**

It was hypothesized that the Golgi Apparatus changed orientation relative to the nucleus in response to variable laminar fluid flow. It was theorized that Golgi would orient itself directly on the nucleus, as opposed to being positioned to its side, when seen through a perspective orthogonal to the slide. To this end, I sought to find the level of overlap between the dapi and Golgi layers. This was done trivially, by finding all of the pixels on both the DAPI nuclear and Golgi layers in which the dye/tags exceeded a certain threshold, finding areas where the two layers overlapped, and then dividing that by the total activated DAPI and Golgi count to find the percentage of Nuclear and Golgi Area covered respectively.

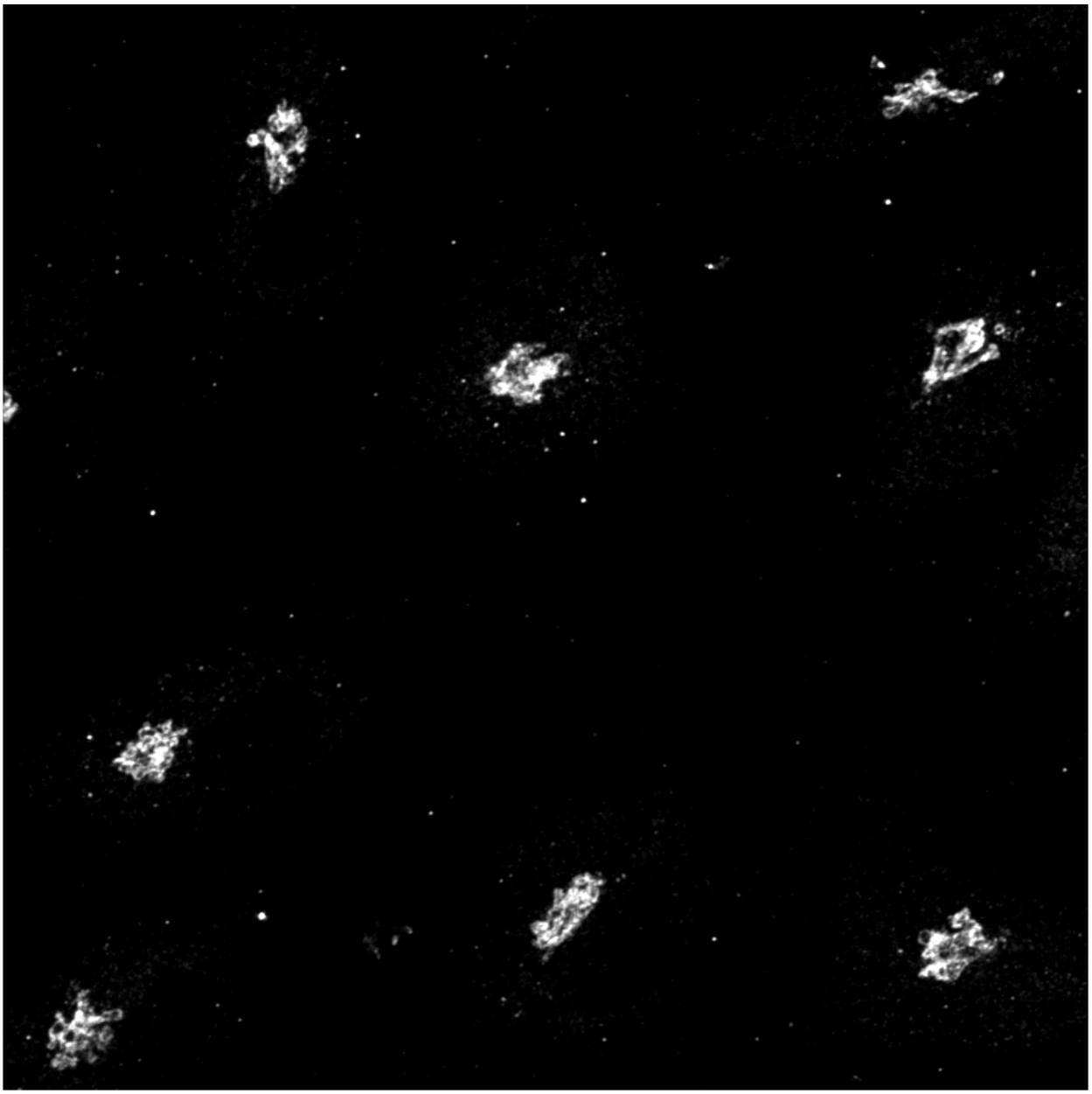

**Figure 6.1: First Input is a .TIFF of the Golgi Apparatus Layer of the Slide**

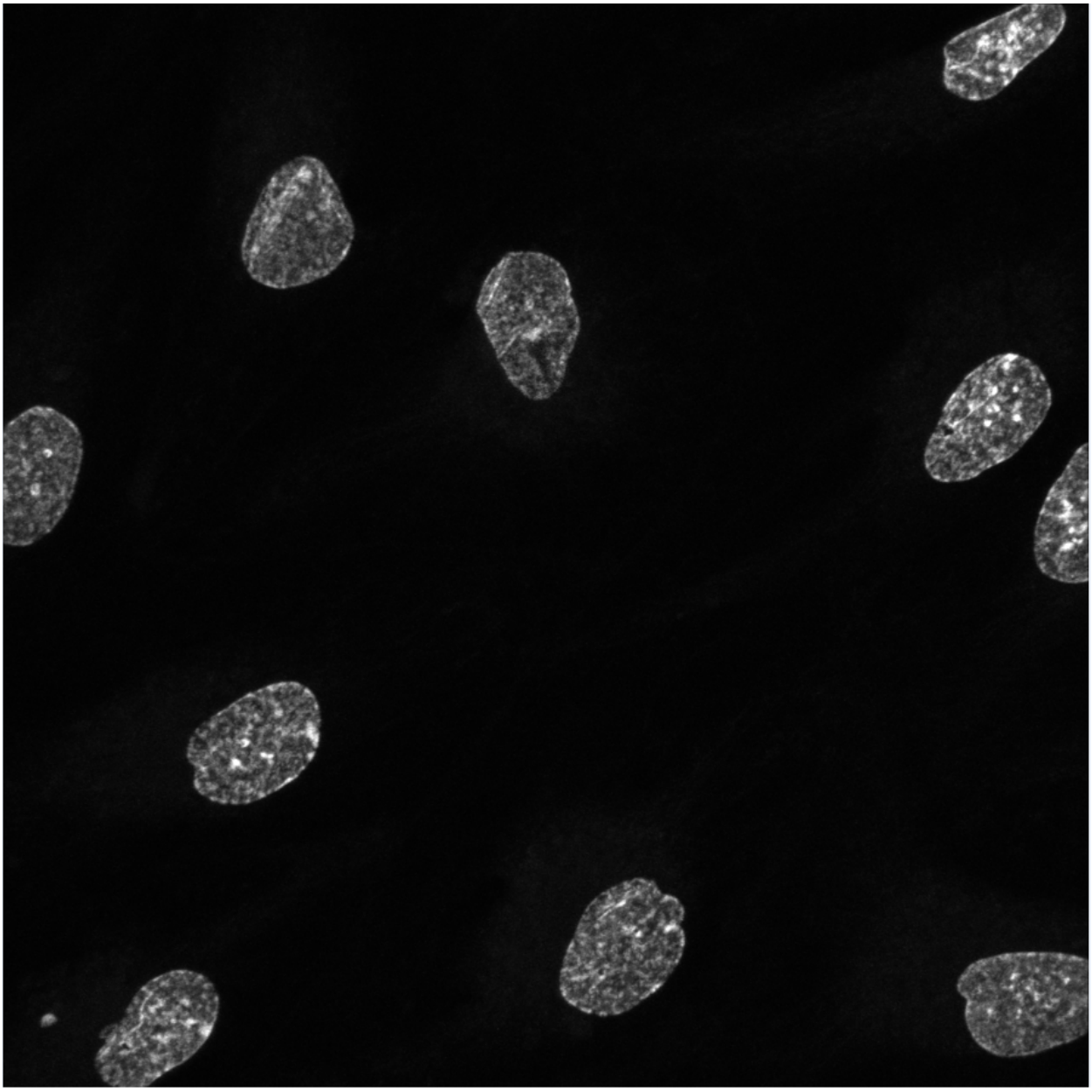

**Figure 6.2: Second Input is a .TIFF of the DAPI-stained nuclear layer of the slide**

The output for this function are two numerical percentage values, instead of some sort of visual output.

**Project #7: Categorizing Golgi morphology orientation and positioning using nuclear structures as reference with respect to the direction of variable laminar fluid flow (the horizontal axis)**

Quantifying and cataloging the positioning, orientation, and general structure of the Golgi between high and low laminar fluid flow was the next step, since the preprocessing of the Golgi slide was complete. The three numerical quantities that were measured for were the tilt of the major axis of the Golgi (if modelled as an elliptical structure), the distance from the center of the Golgi structure to the center of the nucleus, and the ratio of the major axis to the minor axis of the Golgi.

The Golgi was modeled as an ellipse in relation to the nucleus of the corresponding cell using an ad hoc method that gathered tagged points of the Golgi slide that overlapped the perimeter of the nucleus (which could be variably dilated). PCA was then run on that set of points, to create an ellipse that approximated the location, size and orientation of the Golgi Apparatus. The robustness of this approach has not been studied, though, for the given Golgi sample set, it approximated the center of mass, orientation, and distribution of the Golgi sufficiently well enough to draw definitive correlations.

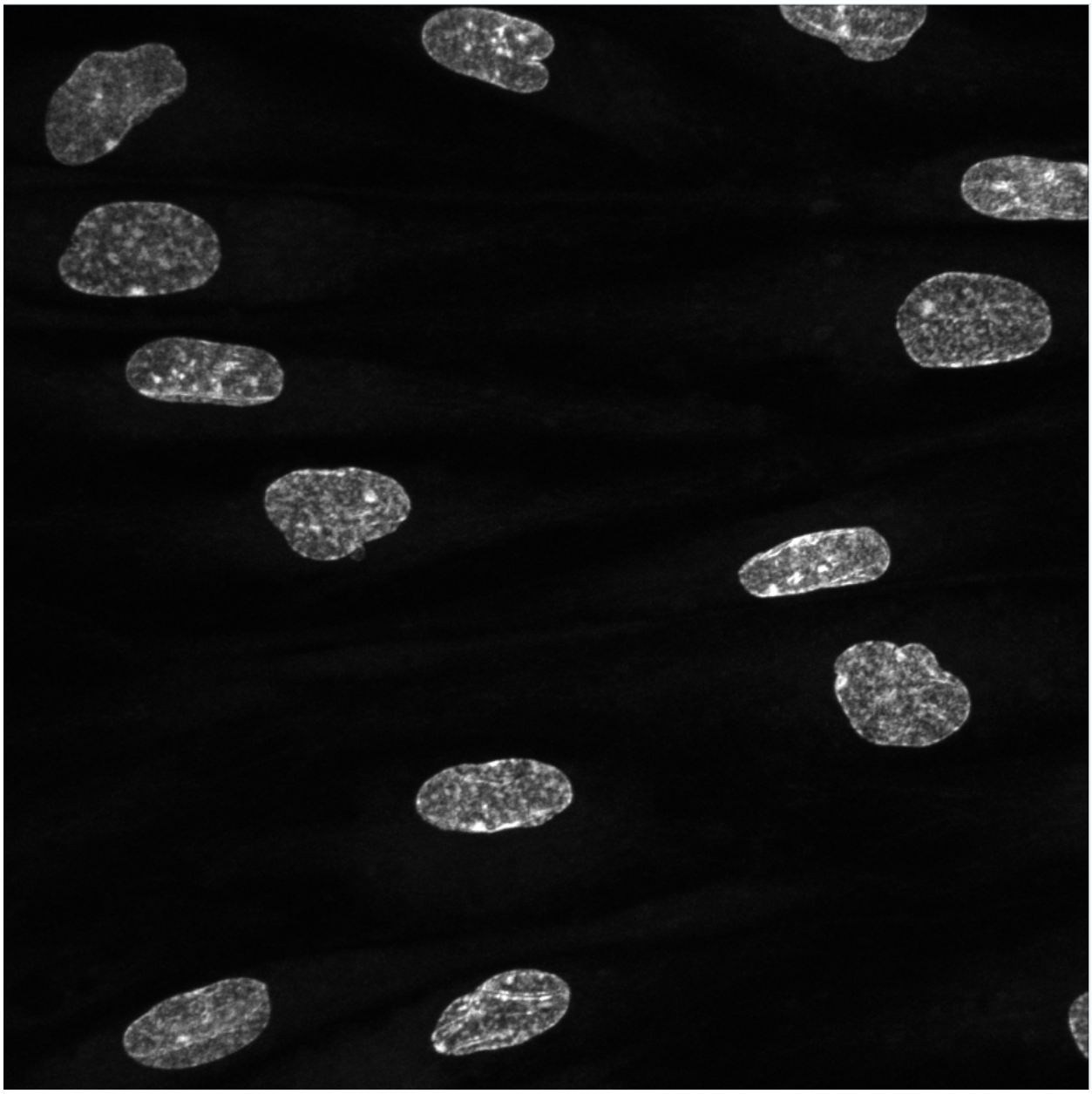

**Figure 7.1: First Input is a .TIFF of the DAPI-stained nuclear layer of the slide**

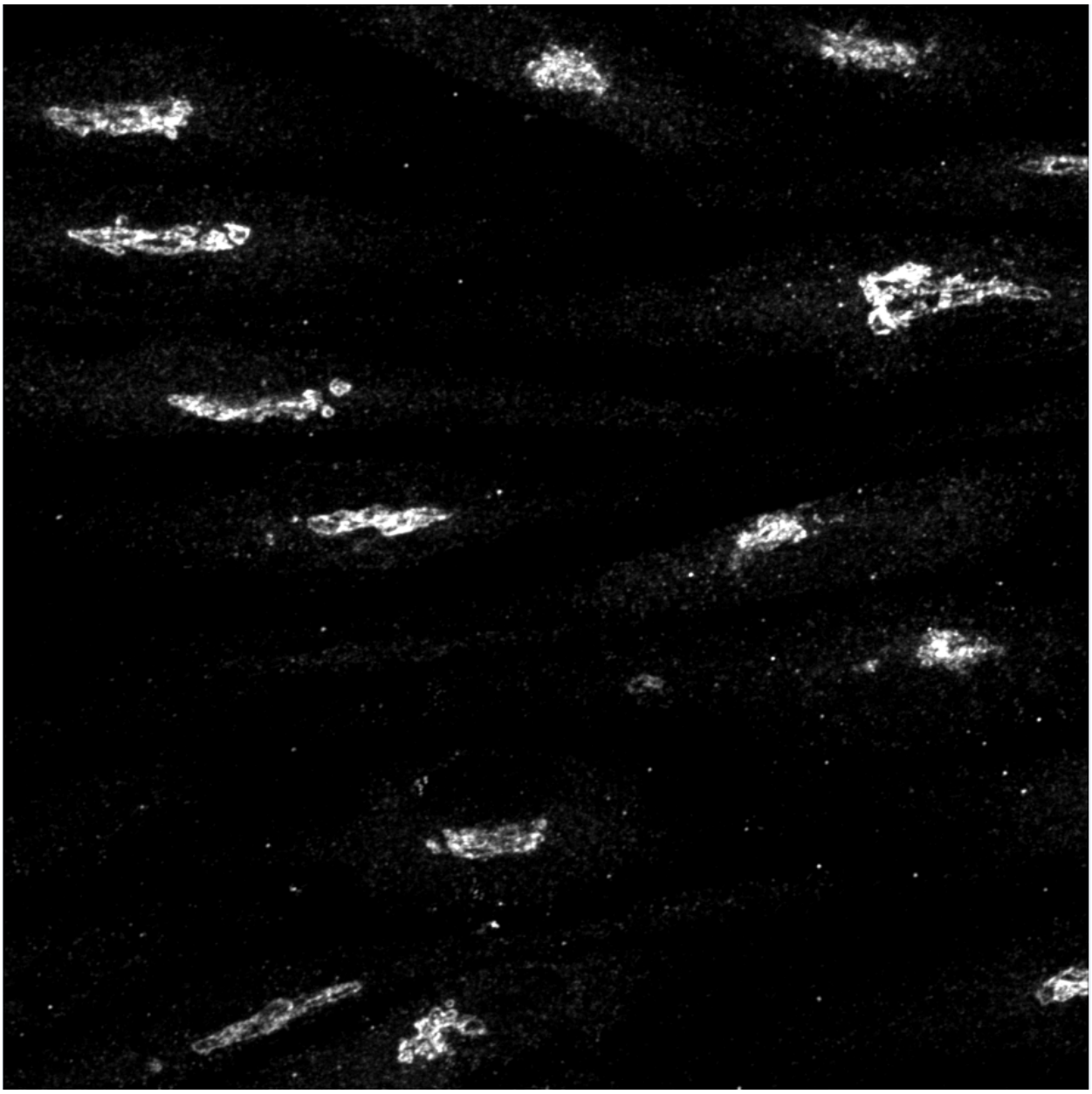

**Figure 7.2: First Input is a .TIFF of the Golgi Apparatus Layer of the Slide**

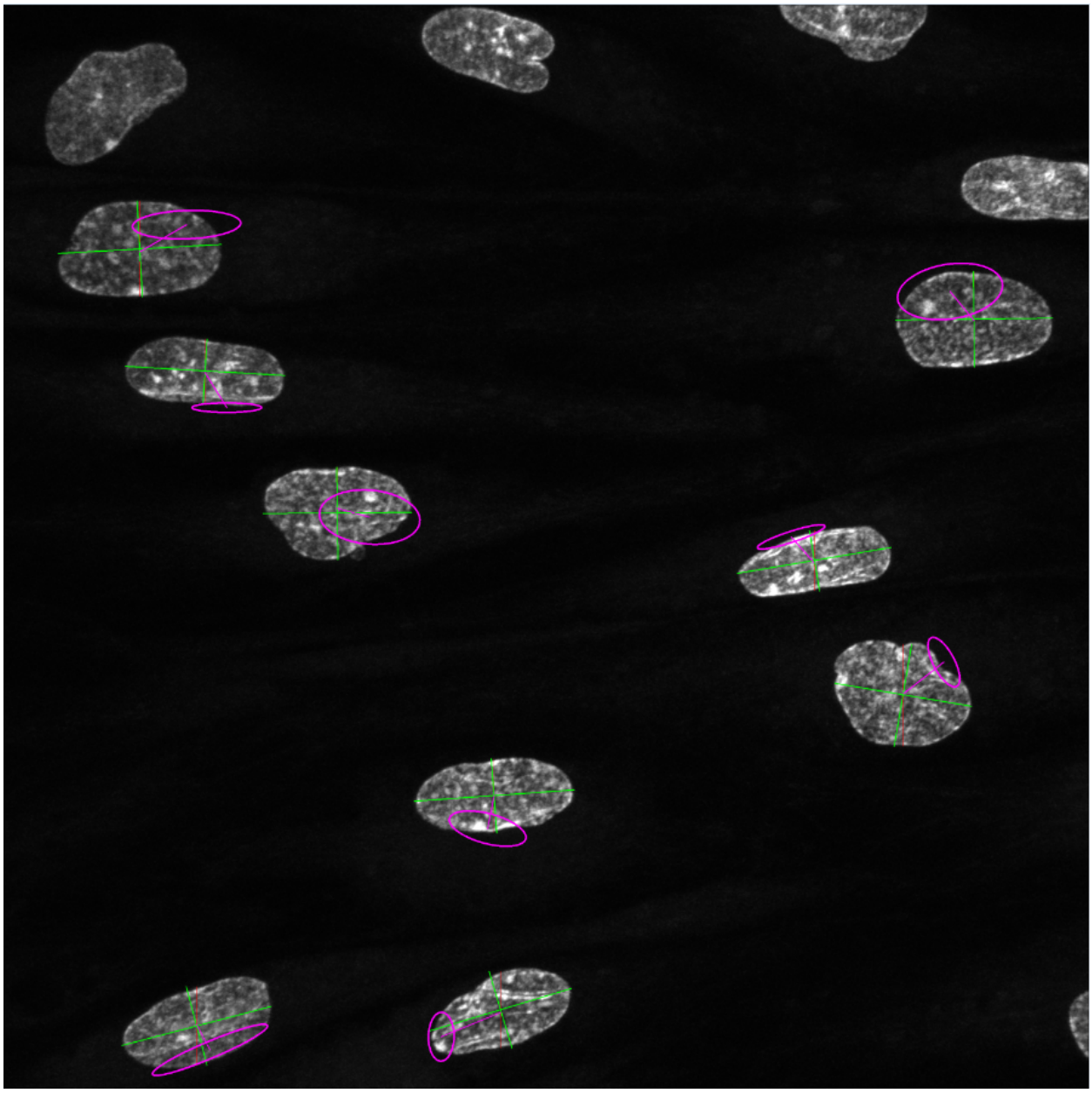

**Figure 7.2: Output is a .TIFF of the PCA of the Golgi over the Nuclear Layer**

**A (non-exhasutive) list of resources**

https://www.sciencedirect.com/science/article/pii/S0006349500765714 [1]

https://docs.opencv.org/3.4/d4/d1b/tutorial_histogram_equalization.html [2]

https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_morphological_ops/py_morphological_ops.html [3]