**Supplemental Methods:**

*Human Cell Sources*

Collection and processing of all human work was done under institutional review board-approved protocol# 11-E1-0245 at NIH. A total of 15 iPSC-RPE cell lines were used in this paper, obtained from 10 different donors. The iPSC-RPE lines were obtained from three types of patients: healthy, AMD patients, and OCA patients. The iPSC-RPE from healthy patients were derived from iPSC lines Healthy-1 and Healthy-2 respectively. The iPSC-RPE from AMD patients were referred to in the paper according to donor number and clone number. For example, AMD1A means the cells came from AMD donor #1 and clone A. The different clones for each donor were replicates, where each clone was completely replicated from generation of iPSCs to iPSC-RPE differentiation. The AMD clones were previously reported (25). A summary of the number of clones per donor is as follows: AMD1 had Clone A and Clone B, AMD2 had Clone A, Clone B, and Clone C, and AMD3 had Clone A, Clone B, and Clone C. The iPSC-RPE obtained from OCA patients (also referred to as albino patients), came from five different patients (a single clone each) and indicated by OCA8, OCA26, OCA103, OCA9, and OCA71. Details on the age and sex of each donor are provided in Table S6.

*Cell Culture Conditions and Media*

All cells were cultured in a cell culture incubator at 37°C and 5% $CO_2$. Depending on the stage of cell development, different cell medium was used. A complete list of the different cell media used in this manuscript are as follows:

Neuro Ectoderm Induction Medium (NEIM) - DMEM/F-12 (Thermo Fisher, 11330-032), KOSR (CTS KnockOut SR XenoFree Kit, Thermo Fisher, A1099201), supplemented with 1% (mass/volume) N-2 (Thermo Fisher, A13707-01), 1x B-27 (Thermo Fisher, 17504-044), LDN-193189 10 μmol/LM (Stemgent, 04-0074-10), SB 431452 10 nmol/L (R&D Systems, 1614), CKI-7 dihydrochloride 0.5 μmol/L (Sigma Aldrich, C0742-5mg), and IGF-1 1 ng/ml (R&D Systems, 291-GMP-5.5ug).

RPE Induction Medium (RPEIM) - DMEM/F-12 (Thermo Fisher, 11330-032), KOSR (Thermo Fisher, A1099201), supplemented with 1% N-2 (Thermo Fisher, A13707-01), 1x B-27 (Thermo Fisher, 17504-044), LDN-193189 100 μmol/L (Stemgent, 04-0074-10), SB 431452 100 nmol/L (R&D Systems, 1614), CKI-7 dihydrochloride 5 μmol/L (Sigma Aldrich, C0742-5mg), IGF-1 10 ng/ml (R&D Systems, 291-GMP-5.5ug), and PD0325901 1 μmol/L (Stemgent 04-0006).

RPE Commitment Medium (RPECM) - DMEM/F-12 (), KOSR (Thermo Fisher, A1099201), supplemented with 1% N-2 (Thermo Fisher, A13707-01), 1x B-27 (Thermo Fisher, 17504-044), nicotinamde 10 mmol/L (Sigma Aldrich, PHR1033-1G), and Activin A 100 ng/ml (R&D Systems, AFL 338).

RPE Growth Medium (RPEGM) – MEM$\alpha$ (Thermo Fisher, 12571-063), supplemented with 1% N-2 (Thermo Fisher, A13707-01), 1% (mass/volume) Glutamine (GlutaMAX Supplement, Thermo Fisher, 35050-061), 1% (mass/volume) non-essential amino acids (MEM Non-Essential Amino Acids Solution (100X), Thermo Fisher, 11140-050), 0.25 mg/ml taurine (Sigma Aldrich, PHR1109-1G), 20 ng/ml hydrocortisone (50 μmol/L solution, Sigma Aldrich, H-6909), 13 pg/ml triiodo-thyronin (Sigma Aldrich, T5516), 5% (volume/volume) FBS (fetal bovine serum, GE Healthcare/Hyclone, SH30071.03).

RPE Maturation Medium (RPEMM) - MEM$\alpha$ (Thermo Fisher, 12571-063), supplemented with 1% (mass/volume) N-2 (Thermo Fisher, A13707-01), 1% (mass/volume) Glutamine (Thermo Fisher, 35050-061), 1% (mass/voume) non-essential amino acids (Thermo Fisher, 11140-050), 0.25 mg/ml taurine (Sigma Aldrich, PHR1109-1G), 10 ng/ml hydrocortisone (Sigma Aldrich, H-6909), 13 pg/ml yriiodo-thyronin (Sigma Aldrich, T5516), 5% (volume/volume) FBS (GE Healthcare/Hyclone, SH30071.03), 50 uM $PGE_2$ (Prostaglandin $E_2$, R&D Systems, 2296).

*Transformation and Differentiation of Human iPSC-RPE*

All iPSC-RPE were developed using a clinical grade protocol that has been outlined previously (25). Briefly, iPSCs clones were derived from CD34+ PBMCs using a previously published protocol (42). iPSCs cells

were then seeded on 5 ug/ml vitronectin (ThermoFisher, A14701S) coated tissue culture plates (Thermo Sci., 140675) or T75 flasks (Corning, 430641U) with E8 Medium (Essential 8 Medium, ThermoFisher, A1517001). After 2 days, cell media was changed to RPEIM for 10 days and then to RPECM for 10 more days. On day 22, cell media was switched to RPEGM. At day 27 cells were disassociated using Versene Solution (0.2 g EDTA (ethylenediaminetetraacetic acid) per liter of phosphate buffered saline, Thermo Fisher, 15040-066) and reseeded in a new culture plate (Thermo Sci., 140675) or T75 flask (Corning, 430641U) with RPEGM. On day 42, cells were disassociated using CTS TrypLE Select Enzyme (Thermo Fisher, A12859-01) and reseeded at 500,000 cells/ml on to biodegradable nanofiber scaffolds [Stellenbosch Nanofiber Company, FiberScaff-RPE$^{TM}$ 3D cell culture scaffolds] (AMD lines) or onto 12-well 0.4 μm polycarbonate transwell plates (Corning, 3401) (Healthy and OCA lines) and cultured with RPEMM. Healthy cell lines were cultured on polycarbonate transwell plates due to the high transparencey of the transwell membrane. AMD lines were cultured on a more clinically relevant biodegradable fibrous membrane, something potentially more difficult to image through. Day 42 seeded RPE were considered to be time 0 iPSC-RPE and all timing for drug treatments, imaging, assay collection, etc. in the experiments outlined in this manuscript count from this day.

Quantitative Bright-field Absorbance Microscopy

*Underlying Principle*

The basic principle behind QBAM imaging was the absorbance measurement, which is an absolute measurement of light attenuation:

$$A(\lambda) = \frac{I(\lambda)}{I_0(\lambda)} \qquad \textit{Equation 1}$$

$A$ is the absorbance value at wavelength $\lambda$, $I(\lambda)$ is the intensity of light passing through a sample and $I0(\lambda)$ is the intensity of light when no sample was present. One reason why absorbance was of interest in this

study was that the Beer-Lambert law establishes the relationship between absorbance and chemical concentration:

$$A(\lambda) = \varepsilon l C \qquad \textit{Equation 2}$$

| | | |
|---|---|---|

$C$ is the chemical concentration in a sample, $\varepsilon$ is the molar attenuation coefficient, and $\ell$ is the path length of the beam of light. In the case of retinal pigment epithelial cells (RPE), melanin is produced by healthy RPE as they mature. Therefore, a doubling of absorbance would indicate a doubling of the amount of melanin within a cell. By converting pixel values in an image of RPE to absorbance values, images become melanin concentration maps that can be tracked over time and non-invasively. However, it should be noted that not all of the signal can be attributed to melanin alone. Scattering effects form the cell membrane and organelles contribute to the absorbance value recorded at a pixel. For the purposes of this paper we assume that this effect is nominal, and all signal is coming from melanin.

Besides the relationship to melanin concentration, there were several advantages to using absorbance imaging but the most significant was reproducibility. Calculation of absorbance values require internal references that make values comparable between microscopes with different configurations. With QBAM imaging, each pixel value was converted to an absorbance value by dividing each pixel's intensity in an image of a sample ($I(\lambda)$) by the corresponding pixel in an image captured when no sample was present ($I0(\lambda)$). As with most measurements, there were several factors to consider when making a measurement to ensure reproducibility. QBAM imaging mitigates some of the sources of error through a variety of procedures including benchmarking, internal calibration, and real time statistics.

*Calculation of Pixel Level Absorbance*

For QBAM imaging, three different images were required to calculate absorbance at every pixel: i) an image captured with the light shutter closed ($IDark(\lambda)$) at exposure time $\epsilon$, ii) an image captured with the light

shutter open without any sample in the field of view ($\mathrm{IBright}(\lambda)$) at exposure time $\epsilon$, and iii) an image of the sample ($\mathrm{I}(\lambda)$). With respect to Equation 1, $\mathrm{IBright}(\lambda)$ was the blank reference image, $\mathrm{I0}(\lambda)$, but an additional term for $\mathrm{IDark}(\lambda)$ needed to be included to account for the camera bias and read current:

$$A(\lambda)_{i,j} = \frac{I(\lambda)_{i,j} - I_{Dark}(\lambda)_{i,j}}{I_{Bright}(\lambda)_{i,j} - I_{Dark}(\lambda)_{i,j}} \qquad \textit{Equation 3}$$

The subscripts i and j indicate pixel positions at the ith row and jth column of each image. Note that the calculation of transmittance (the terms inside of the log function) is the a priori background correction method described by (43). This means that calculation of pixel level absorbance inherently corrects for uneven illumination and was one factor that makes QBAM imaging robust across microscope configurations.

*Calculation of Pixel Level Confidence Intervals*

From Equation 3, there were three different measurements made to calculate absorbance, each with its own potential source of error. To improve reproducibility of pixel level absorbance measurements, our QBAM imaging method uses statistics to calculate confidence intervals for the absorbance value at every pixel. The goal of these statistical methods was to capture sufficient image data to make sure absorbance values at every pixel have a 95% confidence interval of 0.01 absorbance units (10 mAU). This means that the lower end of the dynamic range of QBAM will be 10 mAU.

The standard deviation of absorbance values was calculated as follows:

$$\sigma_{A(\lambda)_{i,j}} = \sqrt{\left(\frac{\sigma_{I(\lambda)_{i,j}}}{I(\lambda)_{i,j} \ln ln\,(10)}\right)^2 + \left(\frac{\sigma_{I_{Bright}(\lambda)_{i,j}}}{I_{Bright}(\lambda)_{i,j} \ln ln\,(10)}\right)^2} \qquad \textit{Equation 4}$$

In Equation 4, $\sigma\mathrm{A}(\lambda)\mathrm{i,j}$ is the standard deviation of the absorbance value at location $(\mathrm{i, j})$. $\sigma\mathrm{I}(\lambda)\mathrm{i,j}$ and $\sigma\mathrm{I}_{\mathrm{Bright}}(\lambda)\mathrm{i,j}$ indicate the standard deviation in pixel intensity values for a pixel at location $(\mathrm{i, j})$ in the sample

image, $I(\lambda)i,j$, and the bright reference image, $I_{Bright}(\lambda)i,j$, respectively. Equation 4 was derived using propagation of uncertainties, and two things should be noted about the derivation. First, it was assumed that $I(\lambda)i,j$ and $I_{Bright}(\lambda)i,j$ were uncorrelated. This was not experimentally verified, but if the variables were correlated then an additional term would be subtracted from Equation 4, which means the standard deviation of absorbance was being overestimated by this formula. Second, we ignore the error introduced by the dark reference image, $I_{Dark}(\lambda)$, because the standard deviation of the dark reference image accounted for less than 1% of the standard deviation of absorbance.

To ensure absorbance measurements were reproducible, a criterion was placed on pixel calculations so that pixel absorbance values have a 95% confidence interval of 0.01 absorbance units. Assuming a normal distribution:

$$0.01 \geq \frac{1.96 * \sigma_{A(\lambda)_{i,j}}}{\sqrt{n}} \qquad \textit{Equation 5}$$

Where $n$ is the number of images captured for $I(\lambda)i,j$. Note that the right-hand side of Equation 5 represents an overestimation of the confidence interval since $n$ is only the images captured of the sample and does not include the number of images captured of $I_{Bright}$.

*Mitigation of Chromatic Aberrations with Color Filters*

When comparing images between different transmitted light microscopes, one reason that images appear to be different is the light spectrum. Everything about how the light spectrum is produced, manipulated with optical components, and collected in the microscope can be different from microscope to microscope. For example, different light sources emit different light spectra that can change with age and temperature, different objectives correct for chromatic aberrations differently, and different cameras have different sensitivities to light of different wavelengths. While the absorbance measurement can mitigate issues associated with spectral emission of the light source and spectral sensitivity of the camera for a single microscope configuration, a color filter is needed to be reproducible between imaging sessions and microscopes. For the current study, images

of fixed RPE (AMD and OCA cells) were captured on a Zeiss AxioImager M2 microscope using three different color filters: 405nm (ET405/10x, Chroma, Bellows Falls, VT), 548nm (ET548/10x, Chroma, Bellows Falls, VT), and 640nm (ET640/20x, Chroma, Bellows Falls, VT). Images of living RPE (Healthy-2 cells) were captured on a Zeiss AxioObserver Z1 using three different color filters: 461 nm (FF01-461/5-25, Semrock, Rochester, NY), 541nm (FF01-541/3-25, Semrock, Rochester, NY), and 671nm (FF01-671/3-25, Semrock, Rochester, NY).

*Microscope Benchmarking and Blank Image Capture*

The purpose of benchmarking is to determine if pixel intensities in microscope images respond linearly to changes in lighting. For QBAM imaging, this is accomplished by finding how pixel intensity changes as camera exposure time is varied, since a doubling of the exposure time should double the pixel intensity. The benchmarking protocol calculates the following quantities: pixel intensity as a function of exposure time, pixel intensity variance as a function of mean pixel intensity, optimal exposure time, and minimum allowed pixel intensity. These quantities and their calculation are described in detail below. This benchmarking protocol is used for each image filter since the image characteristics can change with respect to wavelength. A brief overview of the benchmarking protocol is shown in Figure S1.

The first step of the benchmarking protocol involves capturing images at different exposure times with the camera shutter closed and open. Prior to capturing images, the number of images, n, captured at every exposure time is defined by the user. Then the camera shutter is closed, and n-images were captured at 1ms, 2ms, 4ms, etc, up to 1024ms, and all images were saved for quality control. The camera shutter is then opened, and n-images were captured at 1ms, 2ms, and 4ms. For each set of images captured at one exposure time, the mean and standard deviation of pixel intensity were calculated for every pixel. Then the exposure time is doubled, n-images were captured, and the mean and standard deviation were calculated for every pixel and this process is repeated until the images were overexposed. Overexposure at a particular exposure time was determined by:

$$\sqrt{\sum_{i,j} \frac{\sigma^2_{I(\lambda,\epsilon-1)_{i,j}}}{I*J}} > \sqrt{\sum_{i,j} \frac{\sigma^2_{I(\lambda,\epsilon)_{i,j}}}{I*J}}$$ *Equation 6*

Where $\sigma \mathrm{I}(\lambda,\epsilon)\mathrm{i,j}2$ is the variance in intensity at pixel $(\mathrm{i,j})$ captured at wavelength $\lambda$ and exposure time $2\epsilon$ ms, and $\mathrm{I}*\mathrm{J}$ is the total number of pixels in the image. Since overexposure is defined by reaching the maximum pixel intensity value of the camera, the standard deviation of overexposed pixels will be 0. Therefore, Equation 6 is satisfied when the images captured at exposure time $2\epsilon$ have less variability than the image captured at the previous exposure time $2\epsilon-1$. However, this is a coarse test for overexposure since the standard deviation can fall off sooner than the maximum pixel intensity value (when full well capacity is reached) while still not fitting the criterion for overexposure in Equation 6. For charge coupled devices (CCDs) it is known that pixel intensity follows a Poisson distribution (44), so a function to estimate the standard deviation based on pixel intensity was created:

$$\sigma_\lambda(p) = \alpha\sqrt{p} + \beta$$ *Equation 7*

Where $\sigma\lambda(\mathrm{p})$ is the pixel intensity standard deviation at wavelength, $\lambda$, for pixel intensity, $\mathrm{p}$, and $\alpha$ and $\beta$ were parameters calculated from a linear regression of mean pixel intensities to standard deviations in images captured at 1ms, 2ms, and 4 ms.

A linear regression was then performed to determine how pixel intensity changes with exposure time. Camera exposure times that resulted in overexposed images were excluded from the linear regression, and images were classified as overexposed if images captured at an exposure time had more than 5% error with respect to Equation 7. Following the linear regression of pixel intensity with respect to exposure time, the ideal exposure time was calculated as:

$$\varepsilon_\lambda = \frac{p_D - 3\sigma_\lambda(p_D) - b}{a}$$ *Equation 8*

Where $\epsilon\lambda$ is the ideal exposure time, pD is the maximum camera pixel bit depth, $\sigma\lambda(\mathrm{pD})$ is the standard deviation estimate (from Equation 7), and a and b were the slope and intercept from the linear regression of pixel intensity with respect to exposure time. The ideal exposure time is where the mean exposure time should be three standard deviations below the maximum possible pixel intensity, which minimizes the number of individual overexposed pixels.

Once the ideal exposure time was determined, the number of images to capture and average to obtain IBright was calculated using the standard deviation estimation function from Equation 7:

$$n_{Bright} = \left(\frac{1.96*\sigma_{\lambda}(p_{\epsilon})}{E}\right)^2 \qquad \textit{Equation 9}$$

Where nBright is the number of images required to have a 95% confidence interval of E for the pixel intensity at the ideal exposure, p$\epsilon$. For the present study E was set to 2 pixel intensity units, and this represented ≈0.05% error for the 12-bit cameras used in the study. Setting E=2 allowed an approximation of Equation 9, $\mathrm{n_{Bright}} \approx \sigma\lambda(\mathrm{p}\epsilon)2$, but this approximation should not be used for cameras with a different bit depth.

The minimum pixel intensity ($\mathrm{p_{min}}$) to precisely calculate absorbance was determined. Absorbance is a log-scaled function, and low pixel intensities relative to $\mathrm{I_{Bright}}$ can have much larger errors than higher pixel intensities. During live cell imaging, if any pixels in an image had values smaller than the minimum pixel intensity, then there would be a high likelihood that the absorbance value of that pixel would not have a 95% CI that was less than 0.01. Using Equation 7 and Equation 4, it is possible to estimate the smallest pixel intensity value that satisfies Equation 5:

$$p_{min} = \left(0.01 > \frac{1.96}{\sqrt{n}}\sqrt{\frac{\sigma_{\lambda}(p)}{p*\ln ln\,(10)} + \frac{\sigma_{\lambda}(p_{\epsilon})}{p_{\epsilon}*\ln ln\,(10)}}\right) \qquad \textit{Equation 10}$$

The minimum pixel intensity value is used during live cell imaging to adjust how images were captured to satisfy Equation 5.

*Assessment of Microscope Equilibrium*

For QBAM imaging to work, a microscope must be capable of capturing images in the same field of view with little variability. Multiple factors can alter how images taken at the same field of view could change from one image capture to the next, such as a change in bulb intensity as the bulb heats up or changes in the sensitivity of the camera as it heats up from use. If images were repeatedly captured, then the microscope should reach an equilibrium state where consecutive images have marginal variability between them. The benchmarking methods described above were used to create an equilibrium metric that determined when the microscope reached equilibrium. The equilibrium metric was:

$$Eq = \frac{a_t}{\sum_{i=t-6}^{i=t} \frac{a_i}{7}} \qquad \textit{Equation 11}$$

| | | |
|---|---|---|

Where $\mathrm{Eq}$ is the equilibrium metric, $\mathrm{at}$ is the slope obtained from the linear regression of pixel intensity with respect to exposure time described in the benchmarking section, and $\sum \mathrm{i=t-6i=tai7}$ is the average slope from the current linear regression and the six previous linear regressions. Equation 11 is like the equation for absorbance (Equation 1) except the slope of pixel intensity as a function of exposure time is used to factor in fluctuations in pixel intensity as exposure time changes. The equilibrium metric $\mathrm{Eq}$ should have a value near 0 when the microscope is at equilibrium, and a value larger or smaller than 0 if the microscope is not in equilibrium. To ensure the microscope is in equilibrium, $\mathrm{Eq}$ was calculated at least seven times and the microscope was determined to be in equilibrium is the absolute mean value of $\mathrm{Eq}$ values was less than $5*10^{-5}$ with no individual $\mathrm{Eq}$ value having a magnitude larger than $10^{-4}$. If the first seven $\mathrm{Eq}$ values do not meet the equilibrium criteria, then the older $\mathrm{Eq}$ value is discarded and a new $\mathrm{Eq}$ value is calculated after running the benchmarking protocol again. The process is repeated until the seven most recent $\mathrm{Eq}$ values meet the

equilibrium criteria. We found that these equilibrium criteria were met for microscopes only after the microscope was on for 30 minutes to 1 hour. This protocol was run prior to every live RPE imaging session to ensure microscopes were in equilibrium.

*Sample Imaging and Calculating Absorbance*

After ensuring that the microscope is in equilibrium, the microscope is benchmarked and reference images were captured ($I_{Bright}$ and $I_{Dark}$), and then a sample is imaged. For each field of view of the sample, $n$ images were captured and averaged. The bright reference image was always captured when the microscope was focused on the same medium that the sample was prepared in. For fixed samples of cells mounted on microscope slides, the bright reference image was captured when focused on a blank section of the microscope glass. For live cell imaging, the bright reference image was captured in a well containing the same volume of medium that the cells were cultured in. If the average of any pixel value was less than the minimum pixel value calculated during benchmarking, $pmin$, then the ideal exposure time was doubled and an additional $n$ images were captured and averaged. This process of doubling the exposure time and capturing additional images was repeated until all pixels in the averaged image were larger than $p_{min}$. All images captured at each exposure time were saved and used when converting pixels to absorbance values to increase confidence in the measurement.

To calculate absorbance for each field of view, Equation 3 was used with the mean value of $n$ images used for $I(\lambda)i,j$. For $I(\lambda)i,j<p_{min}$, images captured at a longer exposure time were used to calculate absorbance at the pixel by dividing the pixel intensity by the fold increase in exposure time. Since pixel intensity follows a Poisson distribution, dividing the pixel intensity by the fold increase in exposure time decreases the standard deviation by the same factor resulting in a smaller confidence interval.

Culture, Assays, and Imaging of iPSC-RPE

*Immunostaining*

Tissue preparation for immunohistochemistry was done by placing wells in 4% (mass/volume) paraformaldehyde (Electron Microscopy Science, 157-4-100) for 20 minutes. Immunohistochemistry blocking solution (IBS) consisted of 500 ml of 1x DPBS (Dulbecco's phosphate-buffered saline, Life Technologies, 14190250), 5% (mass/voume) bovine serum albumin (Sigma Aldrich, A3311), 0.5% (mass/volume)Triton X-100 (Sigma Aldrich, X100-100ML), and 0.5% (mass/volume) TWEEN20 (Sigma Aldrich, P2287-100ML). Fixed cells were washed with IBS three times and permeabilized for 2 hours with IBS at room temperature. Cells were then stained with the following primary antibody for 1 hour at room temperature: RPE65 (anti-RPE65 monoclonal antibody [1:300, Abcam, ab13826), Ezrin (monoclonal anti-ezrin antibody, 1:100, Sigma Aldrich, E8897), or GT335 (anti-polyglutamylation modification monoclonal antibody, 1:1000, Adipogen, AG-20B-0020). After primary staining, samples were washed with IBS and a goat anti-mouse IgG Alexa Fluor 594 secondary antibody (1:300, Thermo Fisher, A-11032) was added and allowed to incubate at 4°C overnight. All antibodies were diluted with IBS solution. After overnight incubation, samples were washed with IBS and anti-ZO-1 Alexa Fluor 488 mouse monoclonal antibody (1:200, Thermo Fisher, 339188) was added and allowed to incubate for 1 hour at room temperature. Additionally, nuclei were stained with Hoechst 33342 dye (1:1000, Thermo Fisher, H3570) for 15 minutes at room temperature. After staining cells were washed with D-PBS (Dulbecco's phosphate-buffered saline, Life Technologies, 14190250) and mounted onto slides. All images were captured using a Zeiss AxioImager M2 microscope or Zeiss Axio Scan Z1 slide scanner. Z-stacks were acquired over 50 μm along the z-direction with 1.5 μm steps and the maximum intensity projection was used for all analysis.

*VEGF ELISA Quantification*

Vascular endothelial growth factor (VEGF) was measured from the supernatants collected weekly from each well. At least 5 well replicates were chosen for each treatment at each time and 3 technical replicates per well were measured. ELISAs were performed using a single-plex Luminex® multiplex ELISA following manufacturer's protocols (R&D Systems).

*Electrophysiological Measurements*

Electrical resistance measurements used for prediction of the iPSC-RPE monolayers was measured using an EVOM2 and EndOhm chamber (World Precision Instruments, EVOM2 and ENDOHM-12 respectively). Each line was measured for its resistance across 12 wells per treatment, per week. Intracellular transepithelial potential and resistance measurements in response to 1mM potassium (Sigma Aldrich, P5405-1KG) and 100 μM ATP (Sigma Aldrich, A9187-500MG) in Fig 3d, and Fig S3c were done identically to that of previous reports (33). Briefly, RPE monolayer cultures derived from iPSC lines were mounted on a modified ¨Ussing chamber as described previously (45, 46). Calomel electrodes in series with Ringer's solutions and agar bridges were used to measure the transepithelial potential (TEP). The signals from intracellular microelectrodes were referenced to the basal bath to measure the basolateral membrane potential ($V_b$) and the apical membrane potential ($V_a$) was calculated by the equation: $V_a = V_b$ - TEP. The total transepithelial resistance ($R_t$), and the ratio of the apical to basolateral membrane resistance ($R_A/R_B$) were obtained by passing 2–4 mA current pulses across the tissue and measuring the resultant changes in TEP, $V_a$, and $V_b$.

*Gene Expression of iPSC-RPE*

Total RNA was isolated using NucleoSpin RNA (Machery-Nagel, 740955) per the manufacturer's protocol. RNA was quantified using an ND-1000 spectrophotometer (Nanodrop Technologies) and the manufacturers protocol. cDNA synthesis was performed using the Script cDNA Synthesis Kit (Bio-Rad, 1708891) and the manufacturer provided protocols. Custom made gene arrays containing primers for genes: MITF, PAX6, BEST1, CLDN19, PMEL, TYR, OCA2, RPE65, RLBP1, and housekeeping genes RPL13A and B2M were purchased from Bio-rad. Sybr green based QPCR was run on a ViiA 7 Real-Time PCR System (Thermo Fisher Scientific) using an EXPRESS One-Step SYBR GreenER Kit, with premixed ROX (Thermo Fisher, 1179001K) according to the manufacturer's protocol. Each sample was run in triplicate technical replicates across at least two independent wells (2x HPI4, 3x Aphidicolin, and 5x for control). An HPI4 well had to be excluded due to extremely low RNA extraction from the well and an insufficient quantity of cDNA being

amplified to measure. The average of the $C_T$ of the housekeeping genes for each well were used to determine the $-\Delta C_T$ values plotted. Data was analyzed using R software .

*Electron Microscopy of iPSC-RPE*

Mature iPSC-RPE monolayers were fixed with 10% (mass/volume) glutaraldehyde (Electron Microscopy Sciences, 16120) overnight. Samples were then washed with D-PBS three times and immersed in 25% (mass/volume) ethanol for 10 minutes. The sample was then removed and placed in successive solutions of 50%, 75%, 90%, 95%, and 100% ethanol for 10 minutes each. After 100% ethanol emersion, the samples were removed and placed into a critical point dryer (Leica EM CPD300).) and processed as per the manufacturer's protocol for red blood cells. After CPD processing, samples were gold sputter coated with 10 nm of gold and imaged using a Zeiss EVO 10 scanning electron microscope at a voltage of 10kV and amperage of 10 μA.

*Drug Treatment and Imaging*

Following differentiation, iPSC-RPE from donors Healthy-1 and Healthy-2 were placed on a 12-well 0.4 μm polycarbonate transwell plate (Corning, 3401) at a seeding density of 500,000 cells/ml in 0.5 ml. Only 6 of the 12 wells were used to minimize the time spent outside of the incubator while imaging. QBAM imaging and drug treatments started 2 or 1 weeks after seeding for Healthy-1 and Healthy-2 iPSC-RPE, respectively. For each culture plate, two of the six wells were treated with either 3.0 μmol/L Aphidicolin (Sigma Aldrich, A0781-10MG), 30.0 μM HPI4 (Hedgehog Pathway Inhibitor 4, Sigma Aldrich, H4541-25MG), or no additives to the RPEMM culture medium. RPEMM was changed three times a week and imaged on the same day every week immediately after medium change. To prevent the culture plate lid from fogging during imaging, culture medium in wells were replaced with room temperature RPEMM (with drugs if needed) immediately before imaging since room temperature medium minimized condensation. RPEMM without drug was added to a well without any cells in it to capture the bright reference image for QBAM imaging. For each well, a 4x3 grid of overlapping images were captured (10-15% image overlap) for image stitching using a Zeiss AxioImager M2

microscope with a 10x objective. For Healthy-2 and Healthy-1, a total of 6 plates were prepared, with 6 wells occupied in each plate, yielding 36 total wells imaged and analyzed. The iPSC-RPE from AMD samples were previously reported by Sharma et al (Sharm et al, 2018), and the iPSC-RPE from OCA donors were cultured as described above, but were only imaged after they had matured and were fixed with PFA. In total five well replicates of OCA8, six replicates from OCA9, four replicates of OCA26, eight replicates of OCA71, and four replicates of OCA103 were imaged. For AMD data 1 replicate of AMD1 Clone A, 5 replicates of AMD1 Clone B, 1 Replicate of AMD2 Clone A, 5 replicates of AMD2 Clone B, 1 replicate of AMD2 Clone C, 1 replicate of AMD3 Clone A, 5 replicates of AMD3 Clone B, and 1 replicate of AMD3 Clone C were imaged.

For iPSC-RPE imaged from donor Healthy-2, three color filters were used to capture images, but Healthy-1 were imaged using a different set of filters. The filters used to image Healthy-1 permitted a larger bandwidth of light to pass through them, and had the advantage of shorter exposure times and consequently less time spent imaging cells outside of the incubator. Healthy-1 was imaged first with the broadband filters to ensure that iPSC-RPE maturation was not negatively affected by imaging. Once it was established that Healthy-1 iPSC-RPE maturation was not negatively affected by QBAM imaging, filters with a narrow bandwidth of light were used for imaging of Healthy-2 cells. All above images were acquired using a Zeiss AxioImager M2 microscope.

## Deep Neural Network Prediction of Assays

### *Network Architecture and Training*

A deep convolutional neural network (DNN-F) was designed to predict transepithelial resistance (TER) and VEGF-Ratio. The basic structure of the network was inspired by GoogLeNet (47), consisting of a series of preliminary convolutional layers followed by inception layers deeper in the network. An overview of the network structure is provided in Fig S2. The network takes a 1024x1024x3 image as input and produces two values as output: the TER and VEGF-Ratio values. The network was trained to predict TER values first, then a

VEGF-Ratio prediction layer was added to the end of the network. The three-channel image used as an input was composed of three QBAM images captured at different wavelengths (488nm, 561nm, and 633nm). The network had approximately 11 million parameters.

Prior to training, cell measurements were scaled and weighted to improve the prediction capability of the DNN-F. TER measurements were divided by 1000 and VEGF-Ratio measurements were divided by 10 to roughly scale the values to a range of 0 to 1 for training. When training began, all images were given an equal weight but later in training weights were given to each image based on the TER value associated with the image. The weights were created by binning TER values into bins with a range of 100 Ohms, except the smallest bin contained TER values ranging from 0-135 Ohms. Weights were then assigned by dividing the total number of TER measurements by the number of measurements in each bin. As training continued, the weights were tapered until all weights had a value of 1.

For training, mean squared error (MSE) regression was the objective function. Standard stochastic gradient descent was used with a tapered learning rate starting at 0.001 and tapering to a learning rate of 0.00005. No image pre-processing was performed prior to feeding them into the network, except for random cropping of images. Each image was 1040x1388 pixels, and images in the training data were randomly cropped immediately before feeding into the network while test data were cropped at the same location every time. Test and train data were created by assigning one culture plate as test data and the rest of the plates were used as training data (2 replicates of each treatment per plate; aphidicolin, HPI4, controls). The best network was determined by finding the network with the lowest MSE in the test set.

Fluorescent Image Segmentation

A deep convolutional neural network was designed to segment RPE fluorescently labeled for a tight junction protein (ZO-1), which highlights the cell borders and enables accurate cell segmentation. The purpose of this was to have a highly accurate segmentation method to generate ground truth cell border labels for

QBAM, since cell borders in QBAM images had less contrast than fluorescent images of RPE and made manual segmentation of cell borders more difficult. The approach was to 1) train a DNN (DNN-Z) to segment cell borders in ZO-1 fluorescence images using corresponding images where the cell borders had been drawn in by expert technicians, 2) collect QBAM images and fluorescent images of RPE that had been fluorescently stained for ZO-1, 3) use the DNN-Z to segment cell borders using ZO-1 fluorescence images and 4) use the ZO-1 segmentations to train a new DNN to segment cells in QBAM images (DNN-S).

*Human and Mouse Data*

Images of retinal pigment epithelial (RPE) cells labeled for a cell tight junction protein zonula occluden-1 (ZO-1) were collected from a variety of human donors and under a variety of conditions. Human RPE were derived from induced pluripotent stem cells (iPSC-RPE), and iPSC-RPE were obtained from human donors that were healthy or had one of two different disease phenotypes: oculocutaneous albinism (OCA), or age-related macular degeneration (AMD). Images of iPSC-RPE came from 10 different human donors in total. All iPSC-RPE were imaged after they had matured for at least 6 weeks, and were fixed with paraformaldehyde, labeled for ZO-1 and mounted to a microscope slide. Images were captured on three different microscopes at magnifications ranging from 10x to 40x. The human data was further supplemented with images of ZO-1 labeled whole adult mouse retina labeled for cell borders kindly provided by John Nickerson's laboratory (22).

Reference Segmentation Images and Post-Processing

Among many segmentation methods, we chose the FogBank algorithm (48) to segment each fluorescently labeled image. The FogBank algorithm uses thresholding derived from intensity distributions in combination with geodesic distance maps of edges to establish RPE cell regions. The FogBank segmentation results were reviewed and corrected by human subjects to obtain ground truth segmentation data. After the ground truth data was created, images were split in 256x256 tiles, resulting in 4,064 images that were 256x256 pixels in size.

*Deep Convolutional Neural Network Architecture and Training for RPE Segmentation*

A supervised, DNN based segmentation algorithm was designed in MATLAB (R2017a) with an open-source machine learning framework, MatConvNet (49). The same architecture was used to segment fluorescent images labeled for ZO-1 (cell tight junction stain) and to segment cells in QBAM images. The basic layer structure was the inception layers used in GoogLeNet (47), and the higher order structure followed the U-Net architecture (50). A course structure overview is shown in Fig S2. The network takes a 256x256 pixel image as an input, and outputs a 200x200 pixel image where cell borders have positive pixel values and cell bodies have negative pixel values. The network had more than 15 million parameters.

To train the network, stochastic gradient descent was used with ADADELTA optimization ($\varepsilon=10^{-6}$ and $\rho=0.9$). Batch normalization was used with 10 images in each batch. An F1 score was used to determine the best network model, where correctly labeled cell borders were considered true positives. Prior to the final convolution used to classify pixels, a dropout layer was used with a 50% dropout rate. The objective function was a modified logistic loss function:

$$L(x,c) = B + log(e^{B} + e^{-cx}) \qquad \textit{Equation 12}$$

Where x is the predicted pixel class and c is the actual pixel class with c=+1 representing cell borders and c=-1 representing cell bodies, and $B = max\{0, -cx\}$.

*Image Pre-Processing*

Fluorescent images were pre-processed to normalize pixel values based on region statistics, but QBAM images were not. The normalization process allowed us to 1) scale the images for faster training of the DNN, since scaling image values decreases training time, 2) helps to remove some of the local background fluorescence or improves contrast in poorly stained regions, and 3) allows images of different bit depth or contrast to be processed by the same network. QBAM images were not normalized prior to feeding into the network since the pixel values were on an absolute scale and were generally between 0 and 1. Integral images

were used to calculate the local mean and standard deviation within 127x127 pixel box centered on each pixel in the image. Each pixel was normalized based on the local pixel mean and standard deviation as follows:

$$z_{i,j} = \frac{p_{i,j} - \mu_{i,j}}{\sigma_{i,j}} \qquad \textit{Equation 13}$$

Where $p_{i,j}$ is the pixel value in row i and column j, $\mu_{i,j}$ and $\sigma_{i,j}$ were the mean and standard deviation of the 127x127 pixel region centered on pixel $p_{i,j}$, and $z_{i,j}$ is the normalized pixel value.

*Assignment of Pixel Weights for DNN Training*

To improve the accuracy of the DNN, we adjusted the training weights of each pixel in an image based on normalized pixel values depending on each pixel's classification. This adjustment helped the DNN to correctly label cell borders in regions with poor signal. Cell border pixels were normalized as described by Equation 1, except that $\mu_{i,j}$ and $\sigma_{i,j}$ were the mean and standard deviation of only border pixels in the 127x127 pixel region surrounding $p_{i,j}$. Then, all border pixels with $z_{i,j} > -1$ were assigned a weight of 1, and all other pixels were assigned a weight of $-z_{i,j}$. This causes all bright pixels to have a weight of 1, and dim pixels were assigned a weight that is inversely proportional to the pixel intensity. The weights for cell bodies were assigned based on proximity to the cell border. A distance transform for every cell body was performed with respect to the cell border, and then the square root of the distance was used as the pixel training weight. Cell body weights were trimmed, so that all weights greater than 10 were assigned a value of 10.

Training weights were applied to images during training using two different methods. The first method was the way weights were traditionally applied, where the weights were multiplied by the loss during backpropagation. The second method used the weights as the pixel class labels, c, in the logistic regression function (Equation 1). For both the first and second method, $c > 0$ for cell borders and $c < 0$ for cell bodies. For the first method $c = \pm 1$ and for the second method c was equal to the pixel weight. We found that the second method was better for a couple reasons. First, it yielded a higher F1 score in the test data set. Second, we did

not observe that the F1 scores for the test and train data sets diverged while training the DNN (which would indicate overtraining).

Segmentation of QBAM Images

*Determination of Cell Borders in QBAM Images for DNN Training*

A DNN was developed to segment QBAM images (DNN-S). A subset of the samples were imaged using QBAM imaging in addition to fluorescent imaging of ZO-1, specifically the patients that had AMD. Fluorescent images were captured along with transmitted light images of iPSC-RPE. Then, QBAM imaging was used to capture images of the same region. QBAM images were registered to fluorescent images of iPSC-RPE by finding 256x256 pixel regions that had had at least 97% correlation between the transmitted light images captured during fluorescent imaging and QBAM imaging. Correlation was assessed using generalized normalized cross correlation using the Fast Fourier Transform. The fluorescent images were segmented using the DNN, and the DNN segmentation was used as the cell border label for the QBAM images.

*DNN Ensemble Segmentation of QBAM Images*

Unlike the DNN used to segment fluorescent images of RPE that obtained an F1 score greater than 80%, the DNN to segment QBAM images achieved an F1 score of ≈60%. Part of this appeared to be due to ZO-1 border fluorescence overlapping poorly with the borders observed in the QBAM images in some regions of the RPE monolayer. To improve the segmentation accuracy of the DNN-S, an ensemble approach was used where seven different networks were trained and a consensus of all networks was used to determine RPE borders in QBAM images to achieve an F1 score of 66%.

No preprocessing was performed on QBAM images before they were processed with the DNN-S. The structure of the DNN-S was the same as that used to segment the fluorescent images, except the input image

had three channels instead of one. The three channels were three QBAM images captured of the same field of view, but each QBAM image was captured at a different wavelength (488nm, 561nm, and 633nm).

Segmentation preprocessing for feature extraction

*File selection for Healthy-2 and AMD*

From a directory structure with all red, green and blue bright field images, calibration and raw images, blue-channel derived absorbance images, and ZO-1 fluorescently labelled images. For the Healthy-2 dataset (81,646 Files in 1,564 folders), we selected 2,580 absorbance files of size 1388 pixels x 1040 pixels for segmentation. For the AMD dataset, we chose 20 absorbance images of size 2,700 pixels x 3,000 pixels.

*Invert colors and skeletonize segmentation results*

To remove artifacts and close small gaps a binary morphological operation of closing is applied to segmented images containing cell boundaries. Further on, the segmentations were skeletonized such that the cell boundaries become one pixel thick. Next, the images were inverted so that the foreground corresponds to the interior of cells and the background to the cell border. Finally, the binary inverted images were labelled such as that each cell region was labelled with a unique id.

*Removal of Peripheral Cells in Images*

To compute accurate cell-level features from absorbance images over the reference segmentation masks, each mask was pre-processed to remove the segments that were touching the image border. This pre-processing step eliminates feature values that would correspond to partial cells. The same pre-processing routine was applied to the computed segmentation images when they were evaluated for accuracy of segmentation.

*Image partitioning*

For the AMD dataset, the stitched FOVs into large images pose a scalability challenge on RAM during DNN-based model training. The size of FOVs varies across experimental collections. Thus, we pre-processed collected images by stitching them into a large mosaic image and then partitioning the mosaic image into 512 x 512 image tiles with 0% overlap.

Image feature extraction

*Albino iPSC-RPE feature extraction*

From a directory structure with all red, green and blue bright field images, calibration and raw images, blue-channel derived absorbance images, and ZO-1 fluorescently labelled images with 1,978 Files in 439 folders, we selected 381 blue-channel derived absorbance files for image-level feature extraction. These files correspond to 3 x 3 fields of views (FOVs) with 10% overlap. Due to the fact that some absorbance values were less than zero (calibration artifacts), we created a binary mask per image to eliminate those negative values from feature extraction. A feature vector was computed per absorbance image (FOV) over each corresponding mask using the web image processing pipeline (WIPP) (19). The feature vector contains 14 intensity-based and 5 texture-based image features that were originally implemented in MATLAB (R2017a). The intensity-based features correspond to central moments and entropy characteristics. The texture-based features were derived from gray-level co-occurrence matrix (GLCM) and include contrast, homogeneity, correlation, energy and entropy. The scatter plots of features were generating using a script in R and the ggplot package and selected plots were included in Fig 4 of the main manuscript.

*AMD and Healthy iPSC-RPE Feature Extraction*

In order to extract as many cell-level features as possible, we leveraged multiple widely used image libraries for feature extraction that have been integrated into WIPP (Table S1) (19). While the feature values for the same feature definitions might vary (51), they provide insights on the amount of variation introduced not only by image acquisition but also by image processing. We computed intensity, shape and textural features

using feature extractors in WIPP(19), and specifically, the MATLAB (2017a) and CellProfiler (51) extractors applied to each cell region. Table S1 summarizes these features. Features corresponding to cell regions situated on the borders of the images and to very large connected regions were discarded.

Machine Learning Models of Assay Prediction

Machine learning based regression models were built on cell level features extracted from RPE cells so that the trained models could be used to predict TER and VEGF-Ratio measurements. All the models were tested and trained on identical data sets for comparability. For the prediction of TER for the Healthy donor lines treated with Aphidicolin or HPI4, data was divided into Test and Train data by assigning five culture plates as training data and a sixth as test data (2 replicates of each treatment per plate). A similar procedure was done for VEGF-Ratio predictions except that a subset of wells from plate 5 were chosen to be excluded from the training set and used as the test set (one well from each treatment). This was done because ELISAs were not performed on all well supernatants and thus only a subset of the wells were available for this analysis. The selection of these plates/wells for training/testing was done blind prior to data acquisition or analysis so as to remove any bias in sample selection.

For the AMD cell lines TML models were trained from scratch on different combinations of iPSC-RPE monolayers. A total of 18 unique training image subsets were formed, where each image subset contained test data which had images of one clone from each donor, Table S4 shows all combinations in detail), which were not included in the training data. Average performance across all 18 subsets was used to identify important features regardless of machine learning method used or clone combination used for testing.

*Preprocessing of features*

RPE cell features were extracted per cell. The features include intensity, shape, and textural characteristics. Once extracted the mean, standard deviation, skewness, and kurtosis of each feature for each set of cells within each image partition (described in the “Image Partitioning” Section above) was calculated. It

was found that subsets of features were highly correlated and had varying amplitude ranges. In order to remove highly correlated features and normalize their dynamic range, we performed pre-processing of features. The pre-processing was applied on all features and consists of (1) z-normalization and (2) the Correlation based Feature Selection (CFS) method (52-53) which removed all features with a higher than 99.5% correlation. This was done for all data sets within this manuscript independently. Thus, the highly correlated features within the combined data set of the AMD-iPSC-RPE and Healthy-2 lines used in Fig 6 were analyzed for co-correlation separately from the data set used for AMD-iPSC-RPE sample identity (Fig 7), which was analyzed for co-correlation separately from the Healthy-2 only (Fig 5) data set. This led to different total counts of features used in each of these models. Normalization and removal of highly correlated variables was also performed for the data shown in Fig 4 independently of that shown in all other figures for the training/testing data. Additionally, for Fig 4 z-score regularization and feature correlation were performed on individual cell metrics, not on the mean, standard deviation, skewness, and kurtosis of cells within regions of interest. This was done differently than all other figures in the manuscript because we were interested in assessing DNN-S's ability to accurately segment individual cell features and thus the most granular representation of that data was desired. Thus, the number of metrics for the analysis performed in Fig 4 were much lower than the count for all other models (44 vs ~315).

Among a plethora of ML models for classification and prediction, we considered a subset of ML models that predict a continuous variable, such as TER or VEGF-Ratio measurement. This subset of ML models includes: Multi-Layer Perceptron (MLP), Linear Support Vector Machine (L-SVM), Random Forest (RF), Partial Least Squares Regression (PLSR), and Ridge Regression (RR). Identical models were chosen for donor classification tasks (except for ridge regression which has no classification format). Thus, for classification ridge regression was replaced with a Naïve Bayes (NB) classifier. For all models feature weights were scaled based on their absolute magnitude from 0 to 1 and then averages were taken across all methods to determine which features had the highest relative weight to predictions. Features with high averages indicate that these

important features were consistently identified (i.e., ML model independent) and thus relatively important features for prediction. All models were trained using 10-30 k-fold cross validation and then tested on the "left-out" data that was not within the training or validation sets within the k-fold cross validations.

*Multilayer Perceptron (MLP)*

The MLP model approximates a nonlinear relationship between independent and dependent variables. In a multi-layer perceptron model, cell-level features were considered as independent variables and TER/VEGF values were viewed as dependent variables. We used a MLP model with four-layers (two hidden layers), and 70 neurons used in the first hidden layer and 40 neurons used in the second hidden layer of the model. The number of hidden layers and neurons in each hidden layer were chosen empirically such that a model does not overfit the input data nor was too simple for robustly fitting the input data. To identify the features that were important for TER/VEGF prediction, we used Garson's algorithm(53) to compute the connection weights. The results for MLP modeling are shown in Fig 5 and Fig S6.

*Linear Support Vector Machine (L-SVM)*

Support vector machines construct hyperplanes in a high-dimensional space to maximize separation between data. In the case of linear support vector machines (L-SVMs) the plane is linearly correlated with each dimension/feature in the high dimensional space so that the weights of the L-SVM can be directly translated into weights/importance for features. The SVM function has terms to both optimize the penalization of the hyperplane for the "nearness" of the fit to data (cost) as well as the distance at which a penalization occurs (epsilon). Thus, while training models the cost and epsilon were iterated over a range of 0.001 to 1.2 and 0.001 to 4096, respectively. All model optimization was performed in R using the Liblinear package. Results for L-SVM can be seen in Fig 5 and Fig S6.

*Random Forest (RF)*

Random Forests use “forests” or ensembles of decision trees to predict a factor. A decision tree can be thought of as a flow diagram for decision making. Decision trees are constructed from class-labeled training sequences where each node denotes a test on an attribute and each branch represents the outcome of a test, and each terminal node classifies or regresses. Random forests combine multiple deep decision trees, trained on different parts of the same training set, with the goal of reducing the variance in the decision tree output. As such the number of trees and the depth to which each tree is branched can be optimized. Here 125 trees were assessed across 5 to 1500 branches to determine optimal model performance. All model optimization was performed in R using the Caret package. Relative importance of variables was calculated as shown in . Results for RF can be seen in Fig 5 and Fig S6.

*Partial least squared regression (PLSR)*

PLSR creates a linear regression model by projecting the predicted variables (TER and VEGF-Ratio) and the observable variables (cell image features) to a lower dimensional space. PLSR models find the multidimensional direction in the “X” space (predicted variable) that explains the maximum multidimensional variance direction in the Y space (observable variables). In this way high dimensional data can be used to predict real world outcomes. The only variable to optimize for PLSR is the number of dimensional components needed to predict the desired outcome. For the models optimized in this report components were varied from 1 to 50 to determine which had the highest predictive power. Model optimization was done using the Caret and PLS packages in R. Variable importance was defined as the sum of the absolute values of each feature coefficient within each dimensional component multiplied by the percent of the total variance each component explained of the raw data. Fig 5 and Fig S6

*Ridge regression (RR)*

RR is a specialized form of ordinary least squares regression, where the predictive model is optimized to minimize the sum of squared residuals between the predicted and actual results, that includes a gamma function for removal of colinear, redundant, or confounding variables via optimizing using L2 regularization. As

with ordinary least squares regression variable importance can be determined from the absolute value of the weight of the feature coefficients. Models were optimized to reduce variable numbers to fewer than 25 total features. Model optimization was done using the Caret and foba packages in R.

DNN Models to Classify Donor Identity

A DNN was created that takes two QBAM images of iPSC-RPE and determines if the iPSC-RPE come from the same donor. The input was two 1024x1024x3 pixel images. The basic layer structure consisted of low-rank expansions as described by (54). The low-rank expansion layer consisted of a 1x3 convolutional layer followed by a 3x1 convolutional layer. The first layer had 8 neurons, and each subsequent layer doubled the number of neurons relative to the previous layer. Each convolutional operation was followed by a leaky ReLU layer with a 0.1 leak value. Each low-rank expansion layer was batch normalized and then a 3x3 maximum pooling layer with a stride of 2. With the exception of the first layer, a residual layer was added prior to each maximum pooling operation, where the residual layer consisted of a 1x1 convolutional operation to scale the input layer to the same size as the output layer. The final layer was a 3x3 average pooling layer followed by a fully connected or dense convolutional layer (size = 64x64x256). Stochastic gradient descent with ADADELTA optimization was used using the same settings as used to train the DNN-S. The best network was determined based on an F1 score. Due to the simplicity and small size of this network, we were able to train networks on the same 18 training/testing data sets described in the following section.

Machine Learning Models to Classify Donor Identity and Outliers

Machine learning based classification models were built on cell level features extracted from RPE cells so that the trained models could be used to predict clonal outliers or donor identity. All models were assessed for their ability to classify donor identity correctly except for ridge regression model. In place of ridge

regression, a naïve Bayes model was run. For the prediction of clonal outliers all data was placed into the PCA and clustering of the iPSC-RPE lines using a hierarchical clustering method.

For the AMD cell lines TML models were trained from scratch on different combinations of iPSC-RPE monolayers. A total of 18 unique training image subsets were formed, where each image subset contained test data which had images of one clone from each donor, Table S4 shows all combinations in detail, which were not included in the training data. The approach was to use images of clones from donors to try to predict the "parent" donor line of a clone that the network had never seen before.

*Principle Component Analysis (PCA)*

Identification of developmental outliers was done using PCA and hierarchical clustering. PCA uses an orthogonal transformation to find the hyperplanes of maximum variance best describing the variables by converting a set of correlated variables into a set of dimensionally reduced values of linearly uncorrelated variables called principal components. In large dimensional space (e.g. large number of features) PCA is useful to dimensionally reduce the data to determine if, in aggregate, cell image features can classify different donors/clones from each other. Images were grouped at the clone and day level and the mean, standard deviation, skewness and kurtosis of each feature was calculated for each clone/day combination. These aggregate features were the placed into the PCA in Fig 7a. Principle component 1 and 2 made up more than 75% of the total variance of the data and thus were considered a good indicator of overall sample variability. Importance of features was defined as the sum of the absolute values of each feature coefficient within each dimensional component multiplied by the percent of the total variance each component explained of the raw data. PCA was run using base R.

*Hierarchical clustering*

Hierarchical clustering was done on the output of a PCA in which the mean, standard deviation, skewness, and kurtosis of individual features had been calculated for each clone. For hierarchical clustering

the Euclidean distance between all clones was calculated across all principle components and the complete linkage distance was used for clustering. A split height was chosen at 3 branches to represent the three donors used in the study (indicated as red boxes in Fig 7b). Clustering was done use the base R functions.

*Naïve Bayes*

Naive Bayesian models describes the probability of a cell type being from a donor, based on the prior probability of cells being from that donor with a given set of features. Naive Bayesian models assume strong (naive) independence between the features, i.e. A Naive Bayesian classifier considers each of the cell image features to contribute independently to the probability that a cell is from a given donor regardless of any possible correlations between each feature. Models were optimized for the amount of Laplacian correction needed from 0.0 to 0.1 and whether a normal or kernel density for the features was needed. Feature importance was obtained from the absolute values of the feature coefficients in the best fit model. Model optimization was done using the Caret and klaR packages in R.

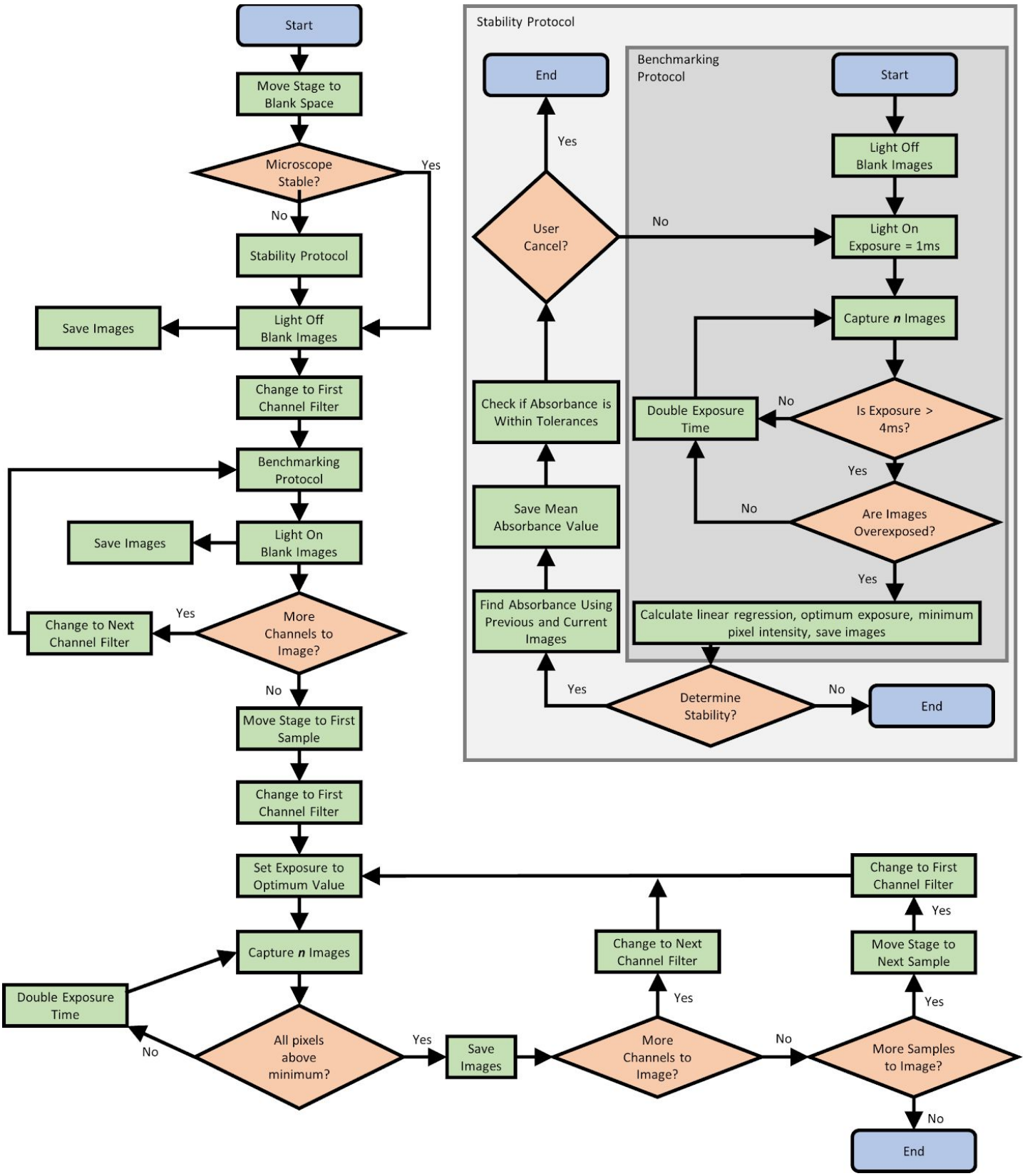



**Figure S1:** Related to Figure 1. A high-level overview of the bright field absorbance imaging protocol.

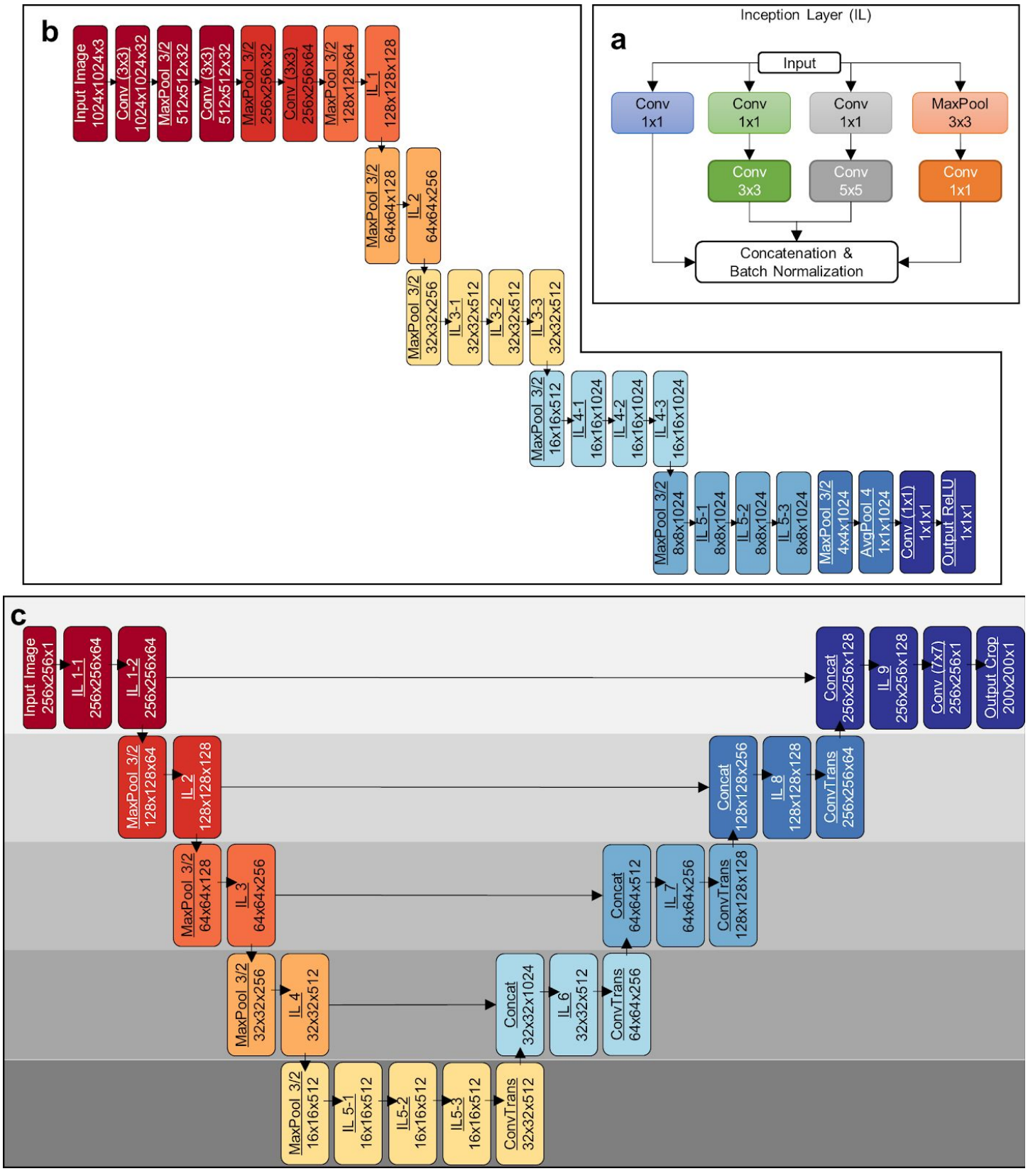

a
Inception Layer (IL)
Input
Conv 1x1
Conv 1x1
Conv 1x1
MaxPool 3x3
Conv 3x3
Conv 5x5
Conv 1x1
Concatenation & Batch Normalization
b
Input Image 1024x1024x3
Conv (3x3) 1024x1024x32
MaxPool 3/2 512x512x32
Conv (3x3) 512x512x32
MaxPool 3/2 256x256x32
Conv (3x3) 256x256x64
MaxPool 3/2 128x128x64
IL 1 128x128x128
MaxPool 3/2 64x64x128
IL 2 64x64x256
MaxPool 3/2 32x32x256
IL 3-1 32x32x512
IL 3-2 32x32x512
IL 3-3 32x32x512
MaxPool 3/2 16x16x512
IL 4-1 16x16x1024
IL 4-2 16x16x1024
IL 4-3 16x16x1024
MaxPool 3/2 8x8x1024
IL 5-1 8x8x1024
IL 5-2 8x8x1024
IL 5-3 8x8x1024
MaxPool 3/2 4x4x1024
AvgPool 4 1x1x1024
Conv (1x1) 1x1x1
Output ReLU 1x1x1
c
Input Image 256x256x1
IL 1-1 256x256x64
IL 1-2 256x256x64
MaxPool 3/2 128x128x64
IL 2 128x128x128
MaxPool 3/2 64x64x128
IL 3 64x64x256
MaxPool 3/2 32x32x256
IL 4 32x32x512
MaxPool 3/2 16x16x512
IL 5-1 16x16x512
IL5-2 16x16x512
IL5-3 16x16x512
ConvTrans 32x32x512
Concat 32x32x1024
IL 6 32x32x512
ConvTrans 64x64x256
Concat 64x64x512
IL 7 64x64x256
ConvTrans 128x128x128
Concat 128x128x256
IL 8 128x128x128
ConvTrans 256x256x64
Concat 256x256x128
IL 9 256x256x128
Conv (7x7) 256x256x1
Output Crop 200x200x1

**Figure S2:** Related to Figure 1. A coarse overview of the DNN architectures used in this manuscript. (a) A GoogLeNet inception layer (IL) was used in both types of DNNs. (b) To predict TER and VEGF-Ratio, a regression DNN (DNN-F) was created where the Output ReLU layer predicted either TER or VEGF. AvgPool4 indicates a 4x4 average pooling, and MaxPool3/2 indicates a window size of 3 and stride of 2. (c) The DNN to segment RPE (DNN-S) was composed of a series of downsizing layers followed by upscaling (convolutional transpose) layers. At each level, downscaling and upscaling layers of the same size are concatenated together. Note: Inception layers on the left side of network did not have a MaxPool component in the inception layer. ReLU was applied after every convolutional operation except the final 7x7 convolution in (c), and the outputs after convolutional operations were padded with 0 to make the outputs the same size as the inputs.

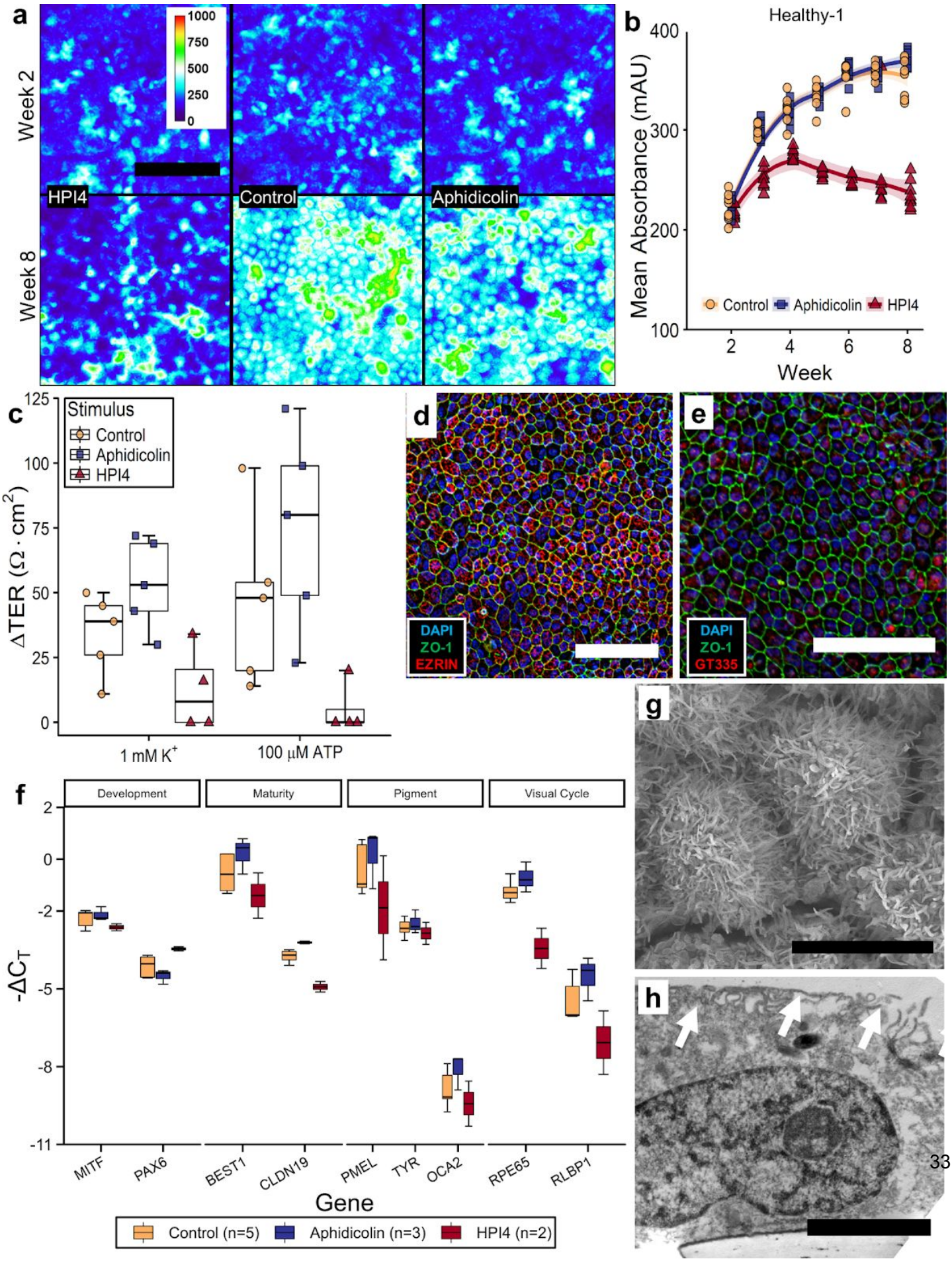
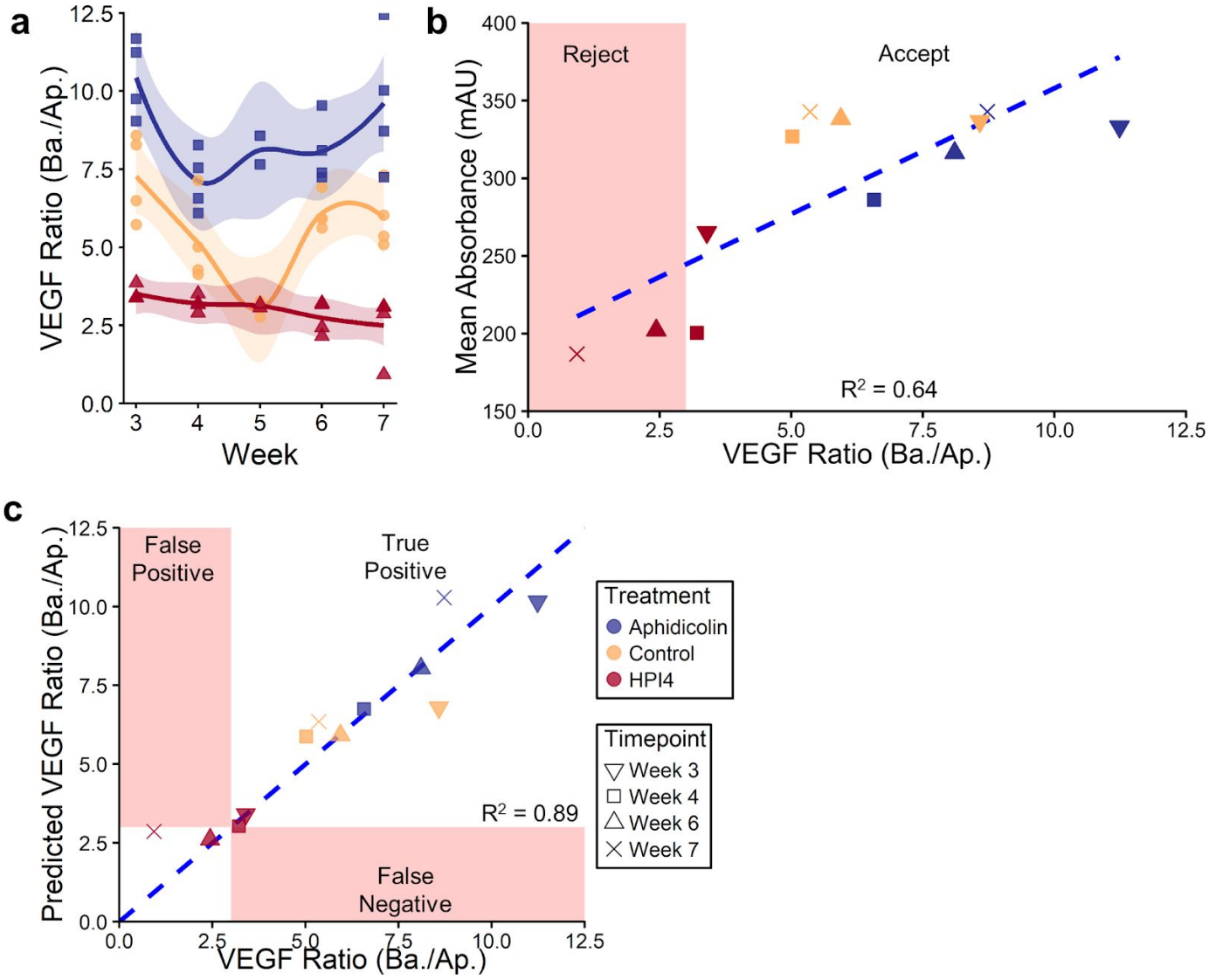

a
Week 2
Week 8
HPI4
Control
Aphidicolin
1000
750
500
250
0
b
Healthy-1
Mean Absorbance (mAU)
Week
Control
Aphidicolin
HPI4
c
Stimulus
Control
Aphidicolin
HPI4
ΔTER (Ω · cm²)
1 mM K+
100 µM ATP
d
DAPI
ZO-1
EZRIN
e
DAPI
ZO-1
GT335
f
Development
Maturity
Pigment
Visual Cycle
-ΔCT
MITF
PAX6
BEST1
CLDN19
PMEL
TYR
OCA2
RPE65
RLBP1
Gene
Control (n=5)
Aphidicolin (n=3)
HPI4 (n=2)
g
h

**Figure S3:** Related to Figure 3. (a) iRPE monolayers from Healthy-1 donor at week 2 of maturation (top row) and week 8 of maturation (bottom row) that were treated with a maturation inhibitor (HPI4), a maturation promoter (Aphidicolin), or neither (control). (b) Mean image absorbance for each well treated over 8-weeks of culture from the Healthy-1 line. (c) Transepithelial resistance (TER) of iRPE monolayers from Healthy-2 donor at week 8 of maturation that were treated with a maturation inhibitor (HPI4), a maturation promoter (Aphidicolin), and neither (control). Each data point in (b) and (c) represents a measurement from one well. (d) Fixed iRPE from Healthy-2 (Control treatment) stained for Ezrin (Red), ZO-1 (Green), and nuclei (Blue) at week 8 of culture. (e) Fixed iRPE from Healthy-2 (Control treatment) stained for GT335 (Red), ZO-1 (Green), and nuclei (Blue) at week 8 of culture. (f) Gene expression measured by PCR of Healthy-2 cells at week 8. Gene expression was measured for eight genes. (g) SEM images of control samples of Healthy-1 iRPE after 8 weeks of culture and imaging showing the development of apical processes. (h) TEM of Healthy-2 RPE showing apical processes as indicated by white arrows. Scale bar for (a, d, and e) is 100 μm, (g) is 10 μm and (h) is 1μm.




**Fig**

**Figure S4:** Related to Fig 3. (a) Healthy-2 VEGF-Ratio in response to Aphidicolin or HPI4 as well as for control cells. The shaded regions represent a 95% confidence interval. (b) The ratio of VEGF on the basal/apical sides of the RPE monolayer plotted against mean QBAM image absorbance. The blue dashed line is linear regression ($R^2$=0.64). (c) A plot of measured VEGF basal/apical ratio versus the DNN-F predicted value. The blue dashed line represents a perfect prediction from the DNN-F ($R^2$=0.89) and has a root mean squared error (RMSE) of 1.1. For biomanufacturing a threshold may be set at a VEGF-ratio of 3.0 (<3.0 reject, ≥3.0 accept) and indicated by the red-shaded areas in (b) and (c). The DNN-F had 100% accuracy, 1.0 sensitivity, and 1.0

specificity in predicting which wells should be accepted/rejected. Each data point in the plots represents one well in a multi-well plate.

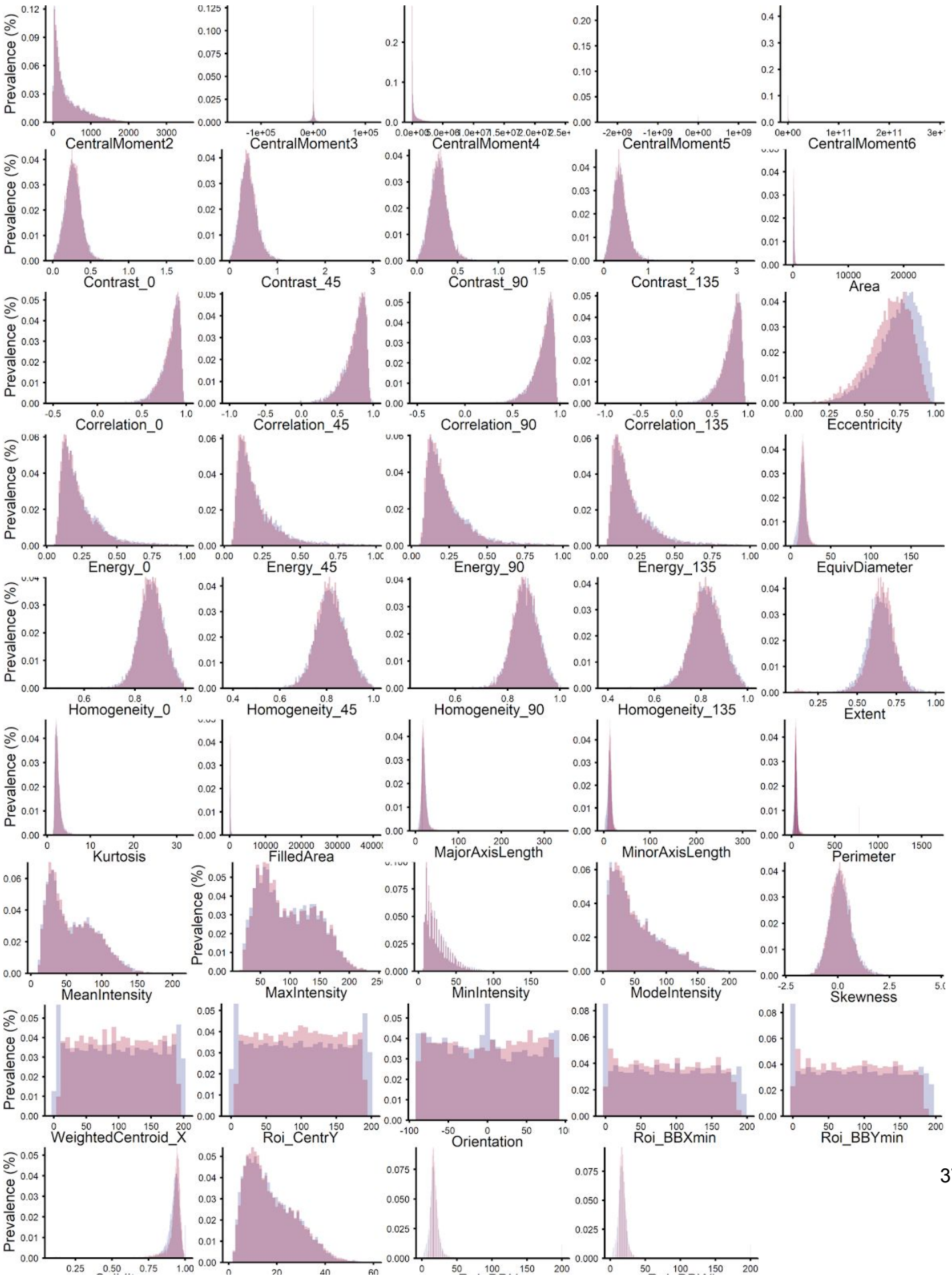

CentralMoment2
CentralMoment3
CentralMoment4
CentralMoment5
CentralMoment6
Contrast_0
Contrast_45
Contrast_90
Contrast_135
Area
Correlation_0
Correlation_45
Correlation_90
Correlation_135
Eccentricity
Energy_0
Energy_45
Energy_90
Energy_135
EquivDiameter
Homogeneity_0
Homogeneity_45
Homogeneity_90
Homogeneity_135
Extent
Kurtosis
FilledArea
MajorAxisLength
MinorAxisLength
Perimeter
MeanIntensity
MaxIntensity
MinIntensity
ModeIntensity
Skewness
WeightedCentroid_X
Roi_CentrY
Orientation
Roi_BBXmin
Roi_BBYmin
Prevalence (%)

**Figure S5:** Related to Fig 5. Comparative histograms of 44 cell image features of RPE segmented by hand (blue) or by DNN-S (Pink). Areas where histograms agree shown as a dark maroon. Each data point in the histograms represents a single cell in one well.

**Figure S6:** Related to Fig 5. Modeling performance of DNN-F and other TML approaches for model TER and VEGF-Ratio data. (a) Summary statistics for TER predictions for DNN-F and all TML algorithms tested. (b) A plot of measured VEGF-Ratio versus the RF predicted value. The black dashed line represents a perfect prediction from the RF. For biomanufacturing a threshold may be set as a ratio of release of 3.0 (<3.0 reject, ≥3.0 accept, indicated by red shading. (c) Summary statistics for VEGF-Ratio predictions for DNN-F and all TML algorithms tested. DNN-F – deep learning network from Fig 2; PLSR – Principle least-squares regression; L-SVM – Linear support vector machine; RR – Ridge regression; RF – Random Forest.

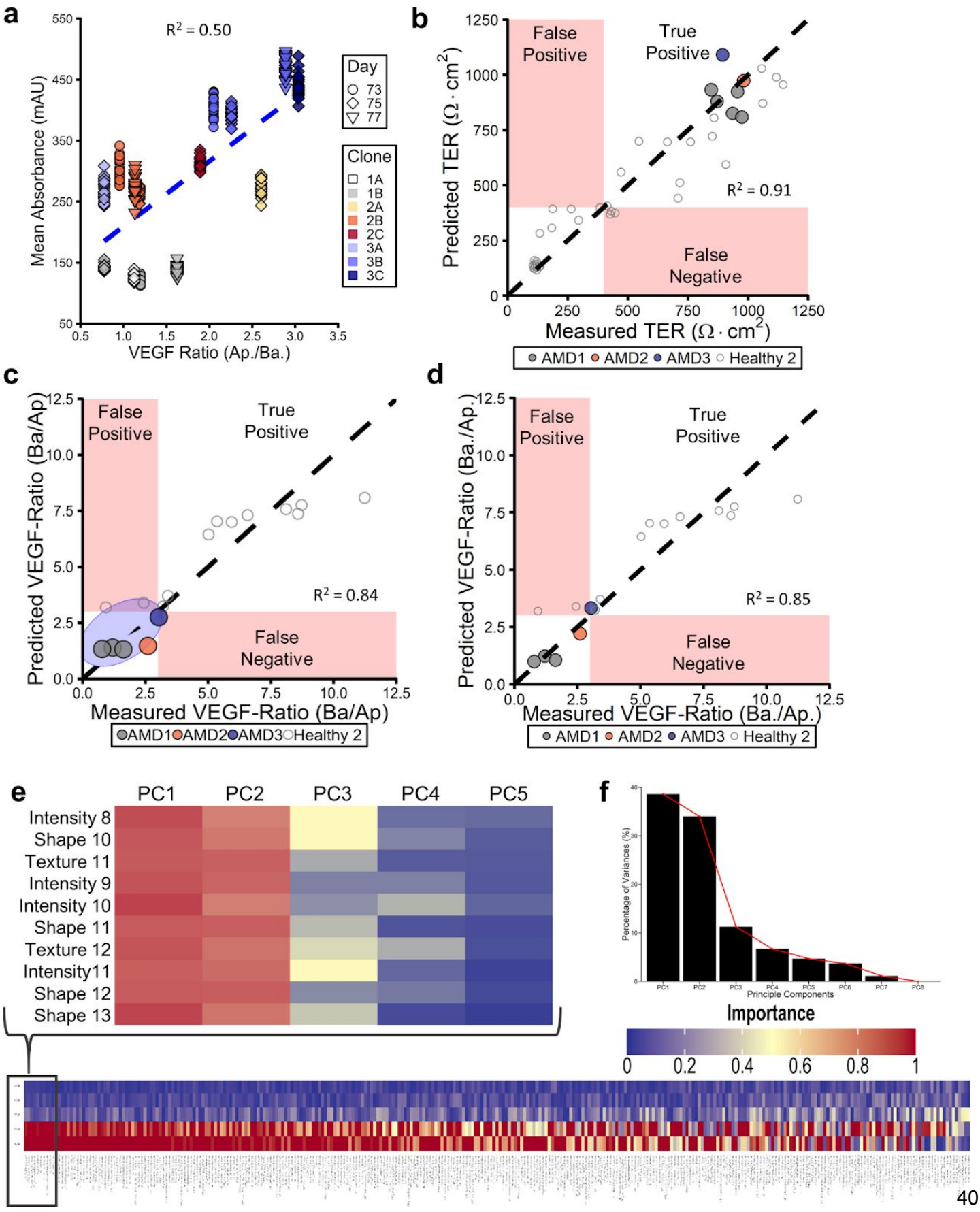

a
R² = 0.50
Mean Absorbance (mAU)
VEGF Ratio (Ap./Ba.)
Day
73
75
77
Clone
1A
1B
2A
2B
2C
3A
3B
3C
b
False Positive
True Positive
False Negative
R² = 0.91
Predicted TER (Ω · cm²)
Measured TER (Ω · cm²)
AMD1 AMD2 AMD3 Healthy 2
c
False Positive
True Positive
False Negative
R² = 0.84
Predicted VEGF-Ratio (Ba/Ap)
Measured VEGF-Ratio (Ba/Ap)
AMD1 AMD2 AMD3 Healthy 2
d
False Positive
True Positive
False Negative
R² = 0.85
Predicted VEGF-Ratio (Ba./Ap.)
Measured VEGF-Ratio (Ba./Ap.)
AMD1 AMD2 AMD3 Healthy 2
e
PC1 PC2 PC3 PC4 PC5
Intensity 8
Shape 10
Texture 11
Intensity 9
Intensity 10
Shape 11
Texture 12
Intensity11
Shape 12
Shape 13
f
Percentage of Variances (%)
Principle Components
Importance
0 0.2 0.4 0.6 0.8 1

**Figure S7:** (a-d) are related to TER and VEGF-Ratio predictions for AMD clones presented in Fig 6 and (e-f) are related to PCA predictions outlier AMD clones presented in Fig 7a,b). TER and VEGF-Ratio predictions of AMD cell line performance. (a) Mean absorbance as assessed by QBAM imaging versus VEGF-Ratio for 8 clones across the last five days of cell maturation, blue dotted line represents the linear best fit for these cells. (b) DNN-F prediction of iRPE TER function across three clones. The black dashed line represents a perfect prediction. (c) Random Forest prediction of iRPE VEGF-Ratio function across three clones that the algorithm had not seen previously. The black dashed line represents a perfect prediction. The light blue ellipse represents the 95% CI of VEGF-Ratio measurements and predictions. (d) DNN-F prediction of iRPE VEGF-Ratio function across three clones that the algorithm had not seen previously. The black dashed line represents a perfect prediction. Red regions in b-d represent the range of values that are classified as out of specification (<400 $\Omega \cdot cm^2$ for b and a VEGF-Ratio of <3 for c and d). (e) The feature importance when AMD clones were analyzed with PCA. Each row is a visual feature and each column is a principle component. All the top five principle components are shown because they make up >90% of the variance of the data, as indicated by (f). Component importance has been weighted by the percent variance that the principle component accounted for and components have been sorted by the mean value across all principle components. To see individual features please refer to the online data. (b) Shows variance accounted for by each PC when aggregated by clone.

## Supplemental Tables

**Table S1:** Related to Fig 1. List of cell feature class types extracted per cell region and the software the feature was derived from. Although the table shows 40 cell features, many features have subsets such as angle or number of moments, yielding a total of 315 cell features that were measured.

| Feature Name | Type | Software |
|---|---|---|
| Eccentricity | Spatial | Matlab |
| Extent | Spatial | Matlab |
| Major Axis Length | Spatial | Matlab |
| Minor Axis Length | Spatial | Matlab |
| Area | Spatial | Matlab |
| Perimeter | Spatial | Matlab |
| Equivalent Diameter | Spatial | Matlab |
| Orientation | Spatial | Matlab |
| Solidity | Spatial | Matlab |
| Euler Number | Spatial | Matlab |
| Filled Area | Spatial | Matlab |
| Convex Area | Spatial | Matlab |
| Mean Intensity | Intensity | Matlab |
| Min Intensity | Intensity | Matlab |
| Max Intensity | Intensity | Matlab |
| Standard Deviation | Intensity | Matlab |
| Contrast | Intensity | Matlab |
| Correlation | Texture | Matlab |
| Energy | Texture | Matlab |
| Homogeneity | Texture | Matlab |
| Median Intensity | Intensity | Matlab |
| Mode Intensity | Intensity | Matlab |
| Skewness | Intensity | Matlab |
| Kurtosis | Intensity | Matlab |
| Entropy | Texture | Matlab |
| Central Moment | Texture | Matlab |
| Number of Neighbors | Spatial | Matlab |
| Haralick Angular Second Moment | Texture | CellProfiler |
| Haralick Contrast | Texture | CellProfiler |
| Haralick Correlation | Texture | CellProfiler |
| Haralick Difference Entropy | Texture | CellProfiler |
| Haralick Entropy | Texture | CellProfiler |
| Haralick Difference Variance | Texture | CellProfiler |
| Haralick Inverse Difference Moment | Texture | CellProfiler |
| Haralick Sum Average | Texture | CellProfiler |
| Haralick Sum Entropy | Texture | CellProfiler |
| Haralick Sum Variance | Texture | CellProfiler |
| Haralick Variance | Texture | CellProfiler |
| Gabor Wavelet features | Texture | CellProfiler |
| Bounding Box Features | Spatial | Java |

**Table S2:** Related to Fig 4. Quantification of the % error and distribution from manual measures of 44 morphological features and from DNN-S segmentations. % Error indicates the absolute difference in counts at

each value in a histogram. KSS, KMMD, and X2 values are three different ways to assess differences in distributions. KSS - two-sample Kolmogorov-Smirnov statistic used in non-parametric distributions. KMMD - Kernel Maximum Mean Discrepancy has no assumptions in initial distribution. X2 is a chi-squared test for normally distributed features.

| Feature | Error (%) | KSS (*p*-Value) | KMMD (*p*-Value) | $X^2$ (*p*-Value) |
|---|---|---|---|---|
| Orientation | 7.7 | <0.001 | 0.150 | 0.247 |
| Bounding Box X Min | 10.1 | <0.001 | 0.190 | 0.624 |
| Bounding Box Y Min | 10.4 | <0.001 | 0.240 | 0.246 |
| Centroid Y | 12.0 | <0.001 | 0.250 | 1.000 |
| Weighted Centroid X | 11.8 | <0.001 | 0.280 | 0.372 |
| Eccentricity | 17.9 | 0.178 | 0.990 | <0.001 |
| Extent | 10.2 | 0.526 | 0.990 | <0.001 |
| Major Axis Length | 9.7 | 0.257 | 0.990 | <0.001 |
| Minor Axis Length | 18.5 | 0.102 | 0.990 | <0.001 |
| Area | 14.2 | 0.514 | 0.990 | <0.001 |
| Perimeter | 12.6 | 0.898 | 0.990 | <0.001 |
| Equivalent Diameter | 13.6 | 0.349 | 0.990 | <0.001 |
| Solidity | 17.1 | 0.007 | 0.990 | <0.001 |
| Filled Area | 14.3 | 0.996 | 0.990 | <0.001 |
| Mean Intensity | 3.3 | 0.174 | 0.990 | <0.001 |
| Min Intensity | 4.6 | 0.178 | 0.990 | <0.001 |
| Max Intensity | 3.4 | 0.003 | 0.990 | 0.256 |
| Standard Deviation | 3.9 | 0.174 | 0.990 | <0.001 |
| Contrast 0 | 5.1 | 0.319 | 0.990 | <0.001 |
| Contrast 45 | 4.0 | 0.531 | 0.990 | <0.001 |
| Contrast 90 | 5.0 | 0.720 | 0.990 | <0.001 |
| Contrast 135 | 4.4 | 0.169 | 0.990 | <0.001 |
| Correlation 0 | 4.6 | 0.007 | 0.990 | <0.001 |
| Correlation 45 | 5.3 | 0.012 | 0.990 | <0.001 |
| Correlation 90 | 4.5 | 0.106 | 0.990 | <0.001 |
| Correlation 135 | 5.0 | 0.178 | 0.990 | <0.001 |
| Energy 0 | 5.0 | 0.066 | 0.990 | <0.001 |
| Energy 45 | 6.1 | 0.066 | 0.990 | <0.001 |
| Energy 90 | 5.2 | 0.066 | 0.990 | <0.001 |
| Energy 135 | 6.2 | 0.066 | 0.990 | <0.001 |
| Homogeneity 0 | 4.6 | 0.175 | 0.990 | <0.001 |
| Homogeneity 45 | 5.3 | 0.144 | 0.990 | <0.001 |
| Homogeneity 90 | 4.7 | 0.330 | 0.990 | <0.001 |
| Homogeneity 135 | 4.2 | 0.077 | 0.990 | <0.001 |
| Mode Intensity | 3.4 | 0.819 | 0.990 | 0.004 |
| Skewness | 4.9 | 0.781 | 0.990 | <0.001 |
| Kurtosis | 5.4 | 0.696 | 0.990 | <0.001 |
| Central Moment 2 | 3.8 | 0.959 | 0.990 | <0.001 |
| Central Moment 3 | 8.0 | 0.283 | 0.990 | <0.001 |
| Central Moment 4 | 4.6 | 0.802 | 0.990 | <0.001 |
| Central Moment 5 | 14.8 | 0.316 | 0.990 | <0.001 |
| Central Moment 6 | 6.6 | 0.985 | 0.990 | <0.001 |
| Bounding Box Width | 12.0 | 0.206 | 0.990 | <0.001 |
| Bounding Box Height | 11.3 | 0.316 | 0.990 | <0.001 |

**Table S3:** Table S3: Related to Fig 5. Subset of the entire list of features for modeling Healthy-2 data with their mean, 95% confidence interval (CI), and Standard Error. The full list can be seen in the online data.

| Week | Treatment | Feature | Class ID | Mean | 95% CI Lower | 95% CI Upper | Std. Error |
|---|---|---|---|---|---|---|---|
| 2 | Aphidicolin | ML_MinIntensity.Mean | Intensity 1 | 144.67 | 142.21 | 147.12 | 1.24 |
| 3 | Aphidicolin | ML_MinIntensity.Mean | Intensity 1 | 204.31 | 200.99 | 207.63 | 1.68 |
| 4 | Aphidicolin | ML_MinIntensity.Mean | Intensity 1 | 146.43 | 144.24 | 148.62 | 1.11 |
| 5 | Aphidicolin | ML_MinIntensity.Mean | Intensity 1 | 183.77 | 179.93 | 187.61 | 1.94 |
| 6 | Aphidicolin | ML_MinIntensity.Mean | Intensity 1 | 185.15 | 182.39 | 187.91 | 1.4 |
| 7 | Aphidicolin | ML_MinIntensity.Mean | Intensity 1 | 180.84 | 177.72 | 183.96 | 1.58 |
| 2 | Control | ML_MinIntensity.Mean | Intensity 1 | 165.16 | 162.11 | 168.21 | 1.54 |
| 3 | Control | ML_MinIntensity.Mean | Intensity 1 | 225.1 | 220.69 | 229.5 | 2.23 |
| 4 | Control | ML_MinIntensity.Mean | Intensity 1 | 187.53 | 183.97 | 191.08 | 1.8 |
| 5 | Control | ML_MinIntensity.Mean | Intensity 1 | 203.86 | 201.35 | 206.36 | 1.27 |
| 6 | Control | ML_MinIntensity.Mean | Intensity 1 | 208.14 | 205.52 | 210.75 | 1.32 |
| 7 | Control | ML_MinIntensity.Mean | Intensity 1 | 213.22 | 210.36 | 216.08 | 1.44 |
| 2 | HPI4 | ML_MinIntensity.Mean | Intensity 1 | 141.81 | 138.76 | 144.85 | 1.54 |
| 3 | HPI4 | ML_MinIntensity.Mean | Intensity 1 | 179.97 | 176.23 | 183.72 | 1.89 |
| 4 | HPI4 | ML_MinIntensity.Mean | Intensity 1 | 104.78 | 102.43 | 107.12 | 1.19 |
| 5 | HPI4 | ML_MinIntensity.Mean | Intensity 1 | 133.78 | 131.13 | 136.42 | 1.34 |
| 6 | HPI4 | ML_MinIntensity.Mean | Intensity 1 | 126.02 | 124.17 | 127.87 | 0.94 |
| 7 | HPI4 | ML_MinIntensity.Mean | Intensity 1 | 110.26 | 108.46 | 112.06 | 0.91 |
| 2 | Aphidicolin | CPI_IntegratedIntensityEdge.Kurtosis | Intensity 10 | 5.87 | 5.33 | 6.41 | 0.27 |
| 3 | Aphidicolin | CPI_IntegratedIntensityEdge.Kurtosis | Intensity 10 | 6.07 | 5.36 | 6.77 | 0.36 |
| 4 | Aphidicolin | CPI_IntegratedIntensityEdge.Kurtosis | Intensity 10 | 5.3 | 4.85 | 5.76 | 0.23 |
| 5 | Aphidicolin | CPI_IntegratedIntensityEdge.Kurtosis | Intensity 10 | 5.59 | 4.79 | 6.38 | 0.4 |
| 6 | Aphidicolin | CPI_IntegratedIntensityEdge.Kurtosis | Intensity 10 | 4.76 | 4.26 | 5.26 | 0.25 |
| 7 | Aphidicolin | CPI_IntegratedIntensityEdge.Kurtosis | Intensity 10 | 4.53 | 3.97 | 5.09 | 0.28 |
| 2 | Control | CPI_IntegratedIntensityEdge.Kurtosis | Intensity 10 | 4.98 | 4.5 | 5.46 | 0.24 |
| 3 | Control | CPI_IntegratedIntensityEdge.Kurtosis | Intensity 10 | 6.13 | 5.59 | 6.68 | 0.27 |
| 4 | Control | CPI_IntegratedIntensityEdge.Kurtosis | Intensity 10 | 5.2 | 4.71 | 5.69 | 0.25 |
| 5 | Control | CPI_IntegratedIntensityEdge.Kurtosis | Intensity 10 | 5.1 | 4.6 | 5.6 | 0.25 |
| 6 | Control | CPI_IntegratedIntensityEdge.Kurtosis | Intensity 10 | 4.76 | 4.13 | 5.38 | 0.31 |
| 7 | Control | CPI_IntegratedIntensityEdge.Kurtosis | Intensity 10 | 5.19 | 4.4 | 5.98 | 0.4 |
| 2 | HPI4 | CPI_IntegratedIntensityEdge.Kurtosis | Intensity 10 | 5.11 | 4.65 | 5.56 | 0.23 |
| 3 | HPI4 | CPI_IntegratedIntensityEdge.Kurtosis | Intensity 10 | 6.45 | 5.89 | 7.01 | 0.29 |
| 4 | HPI4 | CPI_IntegratedIntensityEdge.Kurtosis | Intensity 10 | 6.45 | 5.85 | 7.04 | 0.3 |
| 5 | HPI4 | CPI_IntegratedIntensityEdge.Kurtosis | Intensity 10 | 7.21 | 6.62 | 7.8 | 0.3 |
| 6 | HPI4 | CPI_IntegratedIntensityEdge.Kurtosis | Intensity 10 | 8.06 | 7.28 | 8.84 | 0.39 |
| 7 | HPI4 | CPI_IntegratedIntensityEdge.Kurtosis | Intensity 10 | 8.06 | 7.32 | 8.8 | 0.38 |

**Continued in Online Data**

**Table S4:** Related to Fig 6. List of the 18 different tested clone subsets modeled using RF and the RMSE of TER and VEGF-Ratio for each clone subset. 18 independent models were trained and validated, one for each clone combination listed below.

| Clones Tested | TER RMSE (Ω·cm2) | VEGF-Ratio RMSE |
|---|---|---|
| 1A_2A_3A | 60.7 | 0.98 |
| 1A_2A_3B | 75.5 | 0.66 |
| 1A_2A_3C | 56.5 | 0.69 |
| 1A_2B_3A | 106.8 | 1.29 |
| 1A_2B_3B | 98.4 | 0.92 |
| 1A_2B_3C | 101.2 | 0.77 |
| 1A_2C_3A | 96.1 | 1.09 |
| 1A_2C_3B | 104.1 | 0.58 |
| 1A_2C_3C | 95.2 | 0.38 |
| 1B_2A_3A | 78.9 | 0.76 |
| 1B_2A_3B | 80.3 | 0.61 |
| 1B_2A_3C | 70.9 | 0.60 |
| 1B_2B_3A | 97.2 | 1.25 |
| 1B_2B_3B | 87.7 | 1.14 |
| 1B_2B_3C | 90.2 | 1.19 |
| 1B_2C_3A | 92.6 | 0.92 |
| 1B_2C_3B | 90.2 | 0.52 |
| 1B_2C_3C | 82.3 | 0.36 |
| Mean | 86.9 | 0.82 |
| SD | 14.3 | 0.29 |

**Table S5:** Related to Fig 7c-f. Accuracy of TML algorithms in predicting iRPE donor identity.

| Algorithm | Correct | Incorrect | Total | Accuracy (%) |
|---|---|---|---|---|
| LSVM | 260 | 91 | 351 | 76.4 |
| PLSR | 233 | 97 | 330 | 70.6 |
| RF | 213 | 138 | 351 | 60.7 |
| NB | 210 | 162 | 372 | 56.5 |

**Table S6:** Donor information for cell lines used in this manuscript.

| Cell Line and Donor information | Original Publication | Additional Identifying Information |
|---|---|---|
| Healthy-1 (Age 48, Female) | This paper | Internal Ref: BEST3V |
| Healthy-2 (Age 59, Male) | This paper | Internal Ref: LORDY9 |
| OCA8 (Age 15, Male) | This paper | |
| OCA9 (Age 13, Female) | This paper | |
| OCA26 (Age 14, Female) | This paper | |
| OCA71 (Age 13, Male) | This paper | |
| OCA103 (Age 25, Male) | This paper | |
| AMD1, Clone A (Age 85, Male) | Sharma et al., 2018 | Defined in Sharma et al as D2B |
| AMD1, Clone B (Age 85, Male) | Sharma et al., 2018 | Defined in Sharma et al as D2C |
| AMD2, Clone A (Age 89, Male) | Sharma et al., 2018 | Defined in Sharma et al as D3A |
| AMD2, Clone B (Age 89, Male) | Sharma et al., 2018 | Defined in Sharma et al as D3B |
| AMD2, Clone C (Age 89, Male) | Sharma et al., 2018 | Defined in Sharma et al as D3C |
| AMD3, Clone A (Age 87, Female) | Sharma et al., 2018 | Defined in Sharma et al as D4A |
| AMD3, Clone B (Age 87, Female) | Sharma et al., 2018 | Defined in Sharma et al as D4B |
| AMD3, Clone C (Age 87, Female) | Sharma et al., 2018 | Defined in Sharma et al as D4C |