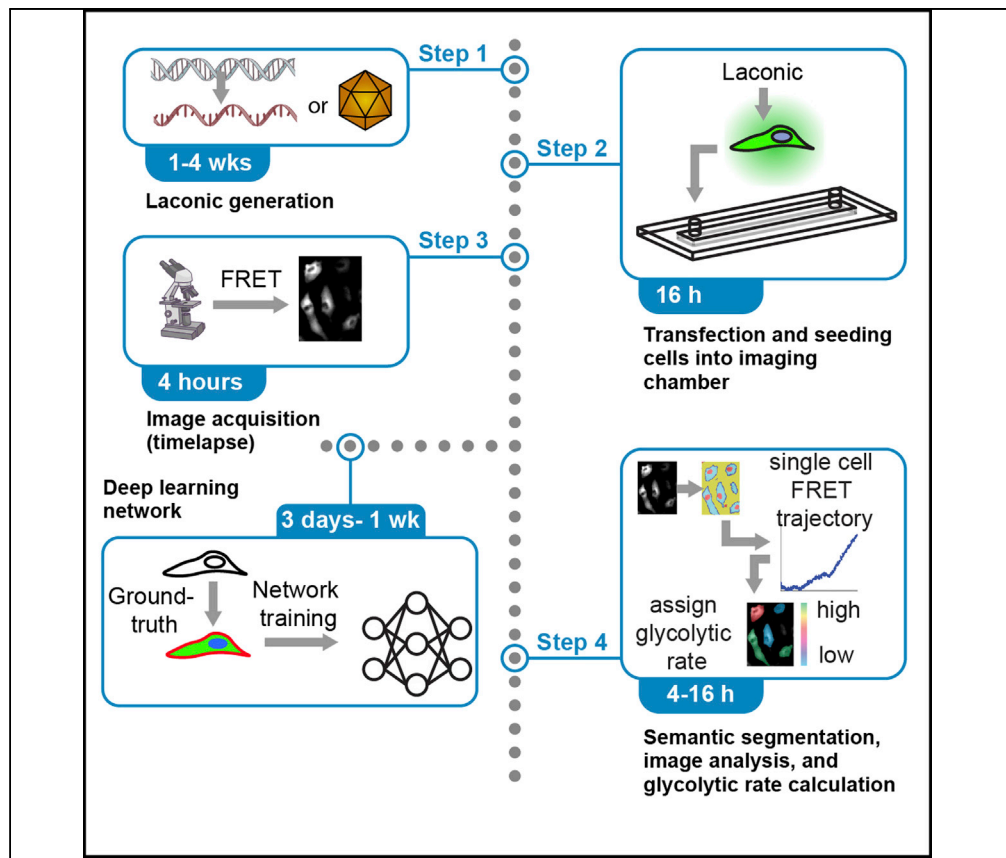


Protocol

Single-cell lactate production rate as a measure of glycolysis in endothelial cells



Heterogeneous metabolism supports critical single-cell functions. Here, we describe deep-learning-enabled image analyses of a genetically-encoded lactate-sensing probe which can accurately quantify metabolite levels and glycolytic rates at the single-cell level. Multiple strategies and test data have been included to obviate possible obstacles including successful sensor expression and accurate segmentation. This protocol reliably discriminates between metabolically diverse subpopulations which can be used to directly link metabolism to functional phenotypes by integrating spatiotemporal information, genetic or pharmacological perturbations, and real-time metabolic states.

Devin Harrison,
David Wu, Jun
Huang, Yun Fang

huangjun@uchicago.edu
(J.H.)
yfang1@medicine.bsd.
uchicago.edu (Y.F.)

Highlights

Protocol for reliable discrimination of heterogeneous single-cell glycolytic states

Multiple strategies for metabolite-sensing probe expression in primary cells

Deep-learning-enabled image segmentation test data and pipeline

Accurate quantification and error estimation of lactate levels and glycolytic rates

Protocol

Single-cell lactate production rate as a measure of glycolysis in endothelial cells

Devin Harrison,^{1,2,3,4} David Wu,^{2,4,5} Jun Huang,^{1,3,*} and Yun Fang^{1,2,6,*}¹Graduate Program in Biophysical Sciences, The University of Chicago, Chicago, IL 60637, USA²Department of Medicine, Biological Sciences Division, The University of Chicago, Chicago, IL 60637, USA³Pritzker School of Molecular Engineering, The University of Chicago, Chicago, IL 60637, USA⁴These authors contributed equally⁵Technical contact⁶Lead contact*Correspondence: huangjun@uchicago.edu (J.H.), yfang1@medicine.bsd.uchicago.edu (Y.F.)
<https://doi.org/10.1016/j.xpro.2021.100807>

SUMMARY

Heterogeneous metabolism supports critical single-cell functions. Here, we describe deep-learning-enabled image analyses of a genetically encoded lactate-sensing probe which can accurately quantify metabolite levels and glycolytic rates at the single-cell level. Multiple strategies and test data have been included to obviate possible obstacles including successful sensor expression and accurate segmentation. This protocol reliably discriminates between metabolically diverse subpopulations which can be used to directly link metabolism to functional phenotypes by integrating spatiotemporal information, genetic or pharmacological perturbations, and real-time metabolic states. For complete details on the use and execution of this protocol, please refer to Wu et al. (2021a).

BEFORE YOU BEGIN

This protocol involves imaging of endothelial cells expressing a genetically encoded sensor for lactate, Laconic (San Martín et al., 2013), within commercially available Ibidi imaging chambers capable of fluid exchange. Laconic is composed of a bacterially-derived lactate binding region flanked by two fluorophores, mTFP and Venus, capable of undergoing fluorescence resonance energy transfer (FRET). In the presence of lactate, the fluorophores will increase in molecular distance, thus reducing FRET. Prior to evaluating the lactate production rate, the sensor, imaging chamber, microscope, and computer for analysis must be set up.

Prepare laconic for genetic introduction to cells

⌚ Timing: 1–4 weeks

Note: Laconic can be introduced to cells using a variety of transfection methods. We found adenoviral transduction into human aortic endothelial cells (HAEC) resulted in reliably homogeneous expression and minimal cell death but have also used mRNA transfection in both primary cells and cell lines. See Figure 1 for comparison of expression via these methods. For generating FRET control proteins, please see Additional Controls.

1. For adenoviral transduction:
 - a. Acquire Laconic plasmid from Addgene (#44238)



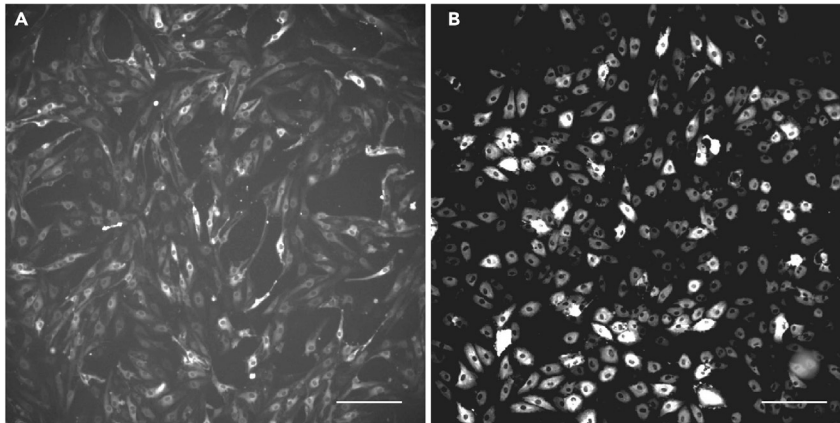


Figure 1. Comparison of different methods of Laconic expression in endothelial cells

Comparison of mRNA versus adenoviral introduction of Laconic into human aortic endothelial cells after 48 h mRNA transfections can sometimes lead to cell elongation whereas transduced cells retain their original cobblestone-like morphology. Scale bars = 200 μ m.

(A) 160 ng/cm² Laconic mRNA transfection.

(B) 50 MOI ad-Lac transduction.

- b. Entrust Vector Biolabs to create Ad5 adenoviral Laconic from the Addgene plasmid ([Link to manufacturer's services](#))
2. For mRNA production:
 - a. Generate template for *in vitro* transcription via PCR using the T7-F and BGH-R primers (see [key resources table](#)) and protocol listed in [Table 1](#) and NEB Q5 Polymerase
 - b. Purify PCR product using QIAquick PCR Purification Kit ([Link to manufacturer's protocol](#))

Note: PCRs can be checked by running on a 1% agarose gel for the correct amplicon size (2403 bp).

- c. Using the PCR-generated template from 2b, create mRNA using mMESSAGING mMACHINE™ T7 ULTRA Transcription Kit ([Link to manufacturer's protocol](#))
- d. After cleaning and eluting mRNA, aliquot (100 ng/ μ L) and store at -80° C up to 6 months for minimal degradation. ([Link to manufacturer's protocol](#))
 - i. When eluting mRNA, preheat the elution solution to 95° C and elute three times from one column using MEGAclean™ Transcription Cleanup Kit.
 - ii. Measure the concentration of the mRNA with a spectrophotometer, such as a BioTek Cytation 3 instrument or NanoDrop, using 1–2 μ L of sample.
 - iii. Concentration of the mRNA can be adjusted with the supplied elution buffer.
 - iv. The lithium chloride precipitation method in the mMESSAGING mMACHINE™ T7 ULTRA Transcription Kit is also sufficient.

Alternatives: In addition to Vector Biolabs, adenoviruses can be generated in house ([Link to useful protocol](#)) and is reviewed in ([Danthinne and Imperiale, 2000](#)). The University of Iowa has a viral vector core facility ([Link to website](#)). Other commercial sources are also available such as Vigene Biosciences ([Link to manufacturer's service](#)) and abm ([Link to manufacturer's service](#)).

Note: Create mRNA transcripts for mTFP and Venus separately (the FRET pair in Laconic) for testing fluorescence bleed through in the filter sets in the microscope (see Additional Controls section).

Table 1. Generating Laconic template for *in vitro* transcription

PCR cycling conditions			
Steps	Temperature	Time	Cycles
Initial Denaturation	98°C	30 s	1
Denaturation	98°C	10 s	25–35 cycles
Annealing	63°C	30 s	
Extension	72°C	1.5 min	
Final Extension	72°C	2 min	1
Hold	4°C	Forever	

Forward (T7-F) and reverse (BGH-R) primers are used in the following PCR reaction in order to generate the template for *in vitro* transcription. PCR reaction steps are based on NEB Q5 Polymerase usage guidelines expecting a 2403 bp amplicon.

Culturing human aortic endothelial cells (HAECs)

⌚ Timing: 5 h

3. Culture HAECs in a 37°C, 5% CO₂ with humidity incubator. HAECs are suitable to use after they have been passaged at least once.
 - a. Pipette 10 mL of EGM2 complete media into a T75 flask and let warm in incubator.
 - b. Rapidly defrost HAECs in a 37°C water bath, shaking.
 - c. Immediately after defrosting, spray the cryo-tube with 70% ethanol and wipe off. In a sterile hood with laminar flow, pipette cells into the pre-warmed T75 flask. Wash the inside of the cryo-tube with media, aspirate, and pipette into T75 flask.
 - d. After 4 h and after cells have adhered to flask, aspirate media and add new, 37°C warmed complete media (to remove DMSO).
 - e. Change media on HAECs every 2–3 days. Split when 80% confluent (see section on plating cells below for further instructions).

Note: HAECs are recommended for use within passages 4–8.

Setup of ibidi imaging chambers

⌚ Timing: 1 h

Ibidi imaging chambers are capable of fluid exchange which will be critical for establishing the calibration curve so that you are able to convert fluorescent values to lactate concentrations. See [Figure 2](#) for assembly.

4. Measure tubing length so that it can be used in your microscope set-up
 - a. Ideally, your microscope will have a chamber for controlling heat, humidity, and CO₂. In order to have fluid exchange without disturbing the chamber, the tubing should be long enough to reach outside of the chamber. The plastic versions of the Ibidi chambers are permeable to gas.
 - b. Test the volume of tubing you have by injecting a known amount of liquid through the line, then measuring how much comes out the other end. This will be important for determining the amount of solution you'll need to inject for thorough washing.
5. Assemble waste/output tubing
 - a. Attach one set of tubing to one elbow male connectors
6. Assemble input/injection tubing
 - a. Attach other set of tubing to one elbow male connector on left side
 - b. Attach female connector on opposite side
 - c. In-line injection port can be attached to female connector
 - d. Rinse thoroughly with 70% ethanol and PBS using syringes

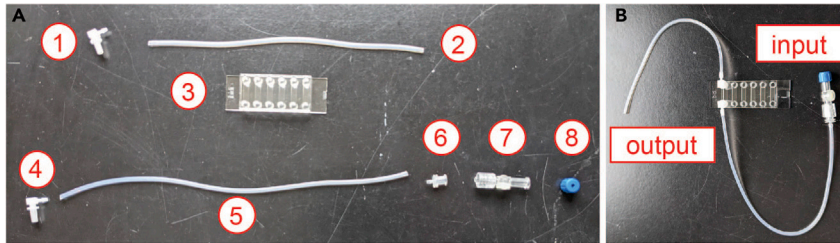


Figure 2. Ibidi imaging chamber assembly

Components of the Ibidi imaging chamber are shown separated in (A) and assembled in (B).

(A) Male connector (1) is attached to tubing (2) to form the output or waste end of the apparatus which will connect to the Ibidi plate (3). Male connector (4) is attached to tubing (5) then to a female connector (6). The in-line injection port (7) is attached to the female end (6) to form the input or injection end of the apparatus. Following rinsing, the cap (8) must be added to close the input port. Once the input and output lines are assembled, they may be reused for multiple Ibidi plates.

(B) When performing imaging experiments, connect the input line before the output line into the bottom and top ports of one well in the Ibidi plate (3).

7. Tubing can be saved for multiple experiments as long as the assembled apparatus is rinsed thoroughly with 70% ethanol and PBS between samples

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Antibodies		
Human VE-Cadherin/CDH5 (D87F2) XP® Rabbit mAb (1:100)	Cell Signaling Technology	Cat#2500S; RRID# AB_10839118
Alexa Fluor 488 Goat Anti-Rabbit IgG (1:1000)	Life Technologies	Cat#A11034; RRID# AB_2576217
Alexa Fluor 594 Goat Anti-Mouse IgG (1:1000)	Life Technologies	Cat#A11037; RRID# AB_2534095
Bacterial and virus strains		
Adenovirus-Laconic	This paper, generated by Vector Biolabs	https://www.vectorbiolabs.com/adenovirus-construction-service/
Chemicals, peptides, and recombinant proteins		
Dimethyloxalylglycine (DMOG)	Sigma-Aldrich	Cat#D3695
4-Chloromercuribenzoic acid (pCMBA)	Sigma-Aldrich	Cat#C5913
Phloretin	Sigma-Aldrich	Cat#P7912
Nigericin	Tocris	Cat#4312
Glucose	Sigma-Aldrich	Cat#G8270
Sodium lactate	Sigma-Aldrich	Cat#L7022-5G
Rotenone	Sigma-Aldrich	Cat#R8875
Sodium chloride (NaCl)	Sigma-Aldrich	Cat#S3014-25KG
Potassium chloride (KCl)	Sigma-Aldrich	Cat#P9333-500G
HEPES (solid)	Thermo Fisher Scientific	Cat#BP310-500
HEPES (solution)	Thermo Fisher Scientific	Cat#15630080
Calcium chloride (CaCl ₂)	Sigma-Aldrich	Cat#C8106
Magnesium sulfate (MgSO ₄)	Sigma-Aldrich	Cat# M7506
Sodium bicarbonate (NaHCO ₃)	Sigma-Aldrich	Cat#S5761-500G
Phosphate buffered saline (PBS), pH 7.4, without Ca ⁺⁺ or Mg ⁺⁺	Invitrogen	Cat#10010
Dulbecco's phosphate buffer saline (DPBS) with Ca ⁺⁺ or Mg ⁺⁺	Invitrogen	Cat#14040
TrypLE express	Gibco	Cat#12605010
Fluorobrite DMEM	Thermo Fisher Scientific	Cat#A1896601
Opti-MEM	Life Technologies	Cat#31985-070
DMEM high glucose	Invitrogen	Cat#NP0366

(Continued on next page)

Continued

REAGENT or RESOURCE	SOURCE	IDENTIFIER
GeneJammer Transfection Reagent	Agilent	Cat#204130
MessengerMAX	Thermo Fisher Scientific	Cat#LMRNA003
EGM2 complete media	Lonza	Cat#CC4176 and CC3156
Paraformaldehyde, 32%	Electron Microscopy Sciences	Cat#15710
Triton-X 100	Sigma-Aldrich	Cat#X100-100ML
Bovine serum albumin	Sigma-Aldrich	Cat#A2153
Tris-buffered saline (TBS), 10×	Sigma-Aldrich	Cat#T5912
Tween-20	Sigma-Aldrich	Cat#P9416
Mounting medium	Thermo Fisher	Cat#P36965
Hoechst33342	Invitrogen	Cat#H3570

Critical commercial assays

mMESSAGE mMACHINE™ T7 ULTRA Transcription Kit	Thermo Fisher Scientific	Cat# AM1344
QIAquick PCR Purification Kit	QIAGEN	Cat#28104
MEGAclear™ Transcription Clean-Up Kit	Thermo Fisher Scientific	Cat# AM1908
NEB Q5® High-Fidelity 2X Master Mix	New England Biolabs	Cat# M04925

Deposited data

Sample data (Wu et al., 2021b)	Zenodo.org	https://doi.org/10.5281/zenodo.4638059
Ground truth sample data set and sample network (Harrison, Devin et al., 2021)	Zenodo.org	https://doi.org/10.5281/zenodo.4898134

Experimental models: cell lines

Human aortic endothelial cells (HAECs)	Lonza	Cat#CC-2535
--	-------	-------------

Oligonucleotides

T7-F, TAATACGACTCACTATAGGG	IDT	N/A
BGH-R, TAGAAGGCACAGTCGAGG	IDT	N/A
mTFPr, TTAGGTACCAGATCTCTTGTA	IDT	N/A
T7-Venus, TTAATATAATACGACTCACTATAGGGACGATTCGATGAAGATCAG	IDT	N/A

Recombinant DNA

Laconic (San Martín et al., 2013)	Addgene	Cat#44238
-----------------------------------	---------	-----------

Software and algorithms

CellProfiler 3.1.8 (Carpenter et al., 2006)	Broad Institute	https://cellprofiler.org/
MATLAB R2018b, including Deep Learning Toolbox	MathWorks	https://www.mathworks.com/products/deep-learning.html
Prism 8	GraphPad	https://www.graphpad.com/scientific-software/prism/
MicroManager	Vale Lab, UCSF	https://micro-manager.org/
MATLAB code and scripts	This paper	https://github.com/wulab-code/STAR_methods

Other

Spectrophotometer	BioTek Cytation 3	Cat#CYT3MF
Olympus IX-71 epifluorescence microscope	Olympus	N/A
Objective (10× Plan C Achromat, NA 0.25)	Olympus	N/A
Excitation filter 438/24	Semrock	Cat#FF02-438/24-25
mTFP emission 483/32	Semrock	Cat#FF01-483/32-25
Venus emission 542/27	Semrock	Cat#FF01-542/27-25
FRET dichroic FF458-Di02	Semrock	Cat#FF458-Di02-25×36
Hoechst excitation 387/50	Semrock	Cat#FF01-387/11-25
Hoechst emission, 447/60	Semrock	Cat#FF02-447/60-25
Hoechst dichroic FF409-Di03	Semrock	Cat#FF409-Di03-25×36
TRITC excitation 534/22	Semrock	Cat#FF01-543/22-25
TRITC emission 593/40	Semrock	Cat#FF01-593/40-25
TRITC dichroic FF562-Di03	Semrock	Cat#FF562-Di03-25×36
Prime 95B	Photometrics	N/A
EXFO X-Cite 120Q	Excelitas	Cat#010-00157

(Continued on next page)

Continued

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Controller for shutter and filter wheel	Prior	Cat#ProscanIII
Automated excitation shutter	Prior	Cat#HF202HT
Automated emission filter wheel	Prior	Cat#HF108IX3
μ-Slide VI Luer, ibidi imaging chambers	ibidi	Cat#80606
5 m Silicone tubing 0.8 mm ID	ibidi	Cat#10841
50 Elbow Luer Connector Male	ibidi	Cat#10802
25 Luer Lock Connector Female	ibidi	Cat#10825
In-line Luer Injection Port	ibidi	Cat#10820
BD needle 25 g 0.62 in	Thermo Fisher Scientific	Cat#10777-019
1 mL Syringes	Thermo Fisher Scientific	Cat#14-829-45
10 mL Syringe	Thermo Fisher Scientific	Cat#14-955-459
75cm ² Tissue Culture Flask	DOT Scientific	Cat# 557341

MATERIALS AND EQUIPMENT

Stock solutions (all filter sterilized)

pCMBA (MCT inhibitor), 500 mM in DMSO, store at –20°C for 1 year

Phloretin (alternative MCT inhibitor for pCMBA), 100 mM in DMSO, store at –20°C for 1 year

Glucose, 1 M in ddH₂O, store at 4°C long term (1 year) or room temperature (20°C–22°C) short term (1 month)

Nigericin, 10 mM in absolute ethanol, store at –20°C for 1 year

Rotenone, 2.5 mM in DMSO, store at –20°C for 1 year

Sodium Lactate, 1 M in water, store at room temperature for 1 year

Additional solutions

TBST. Dilute 10× TBS into ddH₂O, and add 0.1% Tween-20, mix thoroughly. Store at room temperature for 1 year

Paraformaldehyde, 4%. Dilute stock (32%) into PBS, pH 7.4 to make 4% solution before use. Can be frozen at –20°C for up to 12 months.

Blocking buffer: TBST with 3% BSA. Store at 4°C for 6 months.

Viral Media: DMEM high glucose with 10mM HEPES. Store at 4°C for 6 months.

Intracellular Buffer (ICB)

Reagent	Final concentration (mM)
NaCl	10 mM
KCl	130 mM
HEPES	1.25 mM
ddH ₂ O	n/a

Filter sterilize and store at room temperature (20°C–22°C) for 2 years.

Extracellular Buffer (ECB)	
Reagent	Final concentration (mM)
NaCl	112 mM
KCl	5 mM
CaCl ₂	1.25 mM
MgSO ₄	1.25 mM
HEPES	10 mM
NaHCO ₃	24 mM
ddH ₂ O	n/a

pH to 7.4, filter sterilize, and store at room temperature (20°C–22°C) for 2 years.

Alternatives: Phloretin can be used in place of pCMBA for MCT blockade (see Limitations). Trypsin can be used instead of TrypLE.

Optional: Any software can be used to generate figures from analyzed data but we use Prism 8.

Note: ICB and ECB are shelf stable for up to two years but should be checked for formation of precipitate.

Ensure optical setup works with laconic

Laconic is comprised of a bacterial lactate binding domain (LldR) flanked by two fluorophores capable of undergoing Fluorescence resonance energy transfer (FRET). As lactate concentration increases, the FRET ratio between the two fluorophores (mTFP and Venus) is reduced (San Martín et al., 2013). Acquire filter sets that allow for excitation of mTFP and allow for emission in both the donor (mTFP) and acceptor (Venus) channels. Additional sets should be acquired for nuclei and membrane labeling (Hoechst and VE-CADH, respectively) but could also be used for multiplexed imaging.

This experiment requires a standard inverted microscope with either a stage top incubator on top or enclosure. Our heated, humidified, and CO₂ controlled incubator was home built, but commercial ones are available – we recommend Okolab ([Link to manufacturer's website](#)) or Ibidi ([Link to manufacturer's website](#)). No special modifications to the microscope are necessary except for the incubator conditions and automated filter wheel; thus, any inverted microscope could in theory work. We suggest also having a computer-controlled shutter for automatically shuttering the emission light in between frame acquisitions to reduce photobleaching. For acquiring the emission in two channels, an automated emission filter wheel must be used (see [key resources table](#)). We used an epi-illumination metal halide light source. However, an LED light source should also be sufficient. Micro-manager was used to control the automated shutter and filter wheel through a controller, as well as camera.

- Olympus IX-71 epifluorescence microscope with a 10× 0.25 numerical aperture (NA) air objective and Prior automated emission filter wheel and excitation shutter. We chose a 10× objective for maximal data acquisition while still being able to segment the cells. Furthermore, the high depth of field allowed us to capture the fluorescence of the entire cell without resorting to z-stacks.
- FRET filter sets were from Semrock: excitation 438/24, mTFP emission 483/32, Venus emission 542/27, and dichroic FF458-Di02.
- Blue (Hoechst) and red (TRITC) filter sets for visualizing Hoechst stain and VE-CADH were from Semrock: Hoechst excitation 387/50, Hoechst emission, 447/60, Hoechst dichroic FF409-Di03, TRITC excitation 543/22, TRITC emission 593/40, TRITC dichroic FF562-Di03.
- Illumination: EXFO X-Cite metal halide light source

- Camera: Photometrics Prime95B. CMOS camera with high sensitivity, dynamic range, as well as speed. The protocol will likely work with other cameras – integration times may vary.

Alternatives: Instead of using an automated emission filter wheel, split emission systems such as the Optisplit II by Cairn ([Link to Manufacturer's website](#)) can be employed for simultaneous acquisition of both emission channels, although this will change the downstream analysis considerably.

△ **CRITICAL:** Filter sets with very little bleed through are necessary to avoid misinterpretation of FRET. For instance, the excitation in the mTFP channel should minimally excite Venus. The filter sets specified in this protocol minimize bleed through using the excitation light source specified; inclusion of bleed through did not change the calculation of FRET. Changing the light source may change the bleed through characteristics (see Additional Controls).

Optional: Additional filter sets such as those used for TRITC, are not necessary but increase the potential for multiplexed imaging.

Deep learning computer

For training of the segmentation network, a computer outfitted with GPUs will drastically improve training time and segmentation. The computer should also have enough RAM for holding batches of images in memory when performing training. A minimum of 64 GB of memory is recommended. A CPU that has multiple cores is also critical for downstream analysis. For a quick comparison of CPUs check [Geekbench.com](#). Pcpartpicker.com also has useful guides and compatibility charts for parts for building one's own computer, as the pricing and availability of PC parts rapidly evolves. Computers sufficient to perform relatively quick computations are also available from retailers such as Lamda Labs, Bizon Tech, or Nvidia.

- We used 2 NVIDIA GeForce 1080 Ti and 1 NVIDIA Titan V GPUs.
- Our CPU is an AMD Ryzen Threadripper 12-core processor running at 3.50 GHz built in 2018.

Alternatives: It is possible to perform deep learning on cloud computing servers readily available in large Universities or private providers such as Amazon Web Services. The code in this document assumes that computation is occurring locally on a Windows PC.

STEP-BY-STEP METHOD DETAILS

Introduce laconic genetically: Day 1

⌚ Timing: 1–1.4 h with overnight (12–24 h) incubation

In order to have cells expressing Laconic at highest levels, introduce laconic adenovirus vector (ad-Lac) 2 days prior to imaging.

1. Plating cells

- Aspirate media from T75 containing HAEC that is ~80–100% confluent and wash flask with warmed, sterile 5 mL PBS and aspirate.
- Add 1 mL TrypLE and incubate in 37°C, 5% CO₂ with humidity incubator for 1–2 min.
- Check cells under microscope. If cells have become rounded, tap flask gently 10–20 times so that they come off. If cells have not become rounded, wait 1 min. Do not exceed 3 min total. Aim for ~80% of cells to detach.
- Add 5 mL EGM2 complete media to flask, washing the bottom of the flask to remove any adherent cells.
- Aspirate cells and spin at 100 g, 5 min at room temperature.

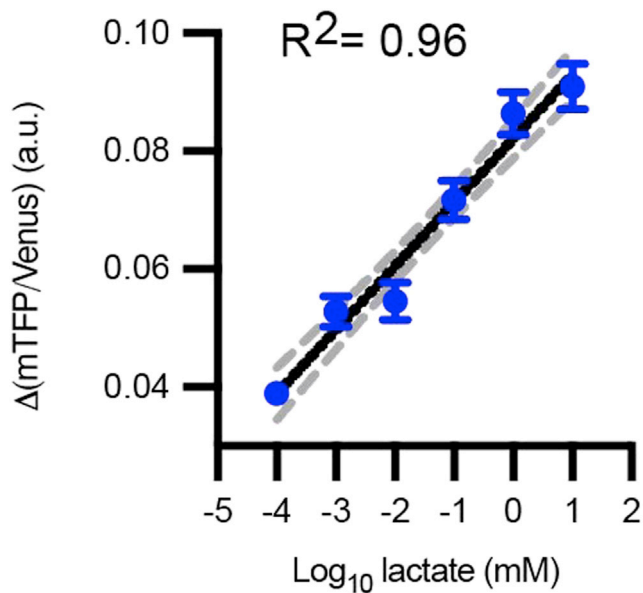


Figure 3. Laconic calibration curve

mTFP/Venus (or 1/FRET) change is linearly correlated over 6 orders of magnitude of log lactate ($n = 287$ cells; black line is semilog fit with $R^2 = 0.96$, gray dotted lines are 95% confidence interval); error bars are SEM. Standard deviation of regression coefficients for R^2 for is 0.049.

- f. Aspirate supernatant and resuspend cells in 1 mL of EGM2 complete media.
- g. Count cells.
- h. Seed cells into wells so that there are $\sim 200,000$ cells/well (in 12 well plate) with enough time before transduction for the cells to adhere (1 day for HAECs).
2. Prepare reagents
 - a. Warm viral media and complete media in 37°C water bath
 - b. Defrost adenovirus laconic stock from -80°C on ice
3. Prepare transduction master mix for half culture volume (500 μL for 12 well)
 - a. Multiplicity of infection (MOI) = Plaque forming units (PFU) of virus used for infection / number of cells. We recommend starting with a MOI of 50:1.
 - i. E.g.: for a virus of $3.40\text{E}+10$ pfu/mL to be used at 50:1 MOI used in 200,000 cells/12 well, use 0.29 μL virus and 1.5 μL GeneJammer in 500 μL viral media per well
 - b. Make master mix if using multiple wells
 - c. Mix by gentle swirling/pipetting. Do not vortex.
4. Aspirate viral media and apply transduction mixture to each well
 - a. Incubate for 1 h in 37°C at 5% CO_2
5. Remove viral media gently and replace with complete media
6. Incubate overnight (12–24 h) at 37°C, 5% CO_2 , humidified.
7. Visualize transfection efficiency on a fluorescence microscope (Figure 1). Transfection efficiency can be roughly estimated by using a microscope and looking for signs of altered cell morphology which signals unhealthy cells. Every cell should be fluorescent with enough signals to obtain an accurate intracellular lactate calibration curve using a reasonable camera integration time (Figure 3). Too much adenovirus (in our hands, 200:1 MOI) results in altered cell morphology and cell death.

△ CRITICAL: Virus must be kept on ice to ensure transduction efficiency. Virus should be aliquoted appropriately so that each tube isn't thawed more than twice.

△ **CRITICAL:** Anything containing virus must be used with caution according to BSL-2 guidelines. Disposables and solutions containing virus must be disposed of separately from other reagents in 10% bleach.

Note: GeneJammer is not necessary but will improve expression.

Note: Solution changes must be done gently following transduction as the cells are especially sensitive to lifting off the plate.

mRNA transfection of laconic (alternative to adenovirus)

⌚ **Timing:** 1 h with overnight (12–24 h) incubation

For mRNA transfection using Lipofectamine™ MessengerMax™ reagent:

8. Prepare cells
 - a. Seed adherent cells into wells so that there are ~200,000 cells/well (in 12 well plate) with enough time before transduction for the cells to adhere (1 day for human aortic endothelial cells). See above on plating cells for directions.
9. Prepare reagents
 - a. Warm Opti-MEM and EGM2 complete media in 37°C water bath
 - b. Defrost mRNA stock from –80°C on ice
10. Make master mix of lipofectamine MessengerMax reagent (Link to manufacturer’s protocol)
 - a. for 12 well plate: 50 µL total solution = 3 µL MessengerMax into 47 µL Opti-MEM.
 - b. Create master mix for multiple wells
 - c. Vortex master mix and incubate at room temperature for 10 min while you prepare mRNA mixes
11. Prepare mRNA solutions so that final concentration is 160 ng/cm²
 - a. For 4 cm² well (12 well), 50 µL solution = 7.1 µL mRNA (100 ng/cm²) into 42.8 µL Opti-MEM
 - b. Mix by triturating
12. Add Master mix of lipofectamine reagent to mRNA solutions in 1:1 volume ratio (50 µL:50 µL for 12 well)
 - a. Mix by triturating
 - b. Incubate at room temperature for 5 min
13. During incubation in step 12b, replace media in wells containing cells to 900 µL complete media total
14. Add 100 µL transfection solution to each well
15. Incubate overnight for a minimum of 12 h up to 24 h.
16. Visualize transfection efficiency on a fluorescence microscope (Figure 1). In HAECs, mRNA transfection is noticeably less efficient than using adenovirus (Figure 1) but still useable from a data acquisition standpoint. Too little mRNA and fluorescence is not enough, but too much mRNA causes cell death.

△ **CRITICAL:** mRNA must be kept on ice to minimize mRNA degradation. mRNA should be aliquoted so that each aliquot isn’t thawed more than twice.

Note: mRNA transfections can be further optimized for your cells by a dose titration. We typically try a range of 25–200 ng/cm².

Plate cells into imaging chambers: Day 2

⌚ **Timing:** 1 h

For ease of solution switching, cells must be plated into Ibidi flow chambers. However, this environment is slightly hypoxic with a smaller volume than traditional cell culture plates so cells must have media changes every 24 h (Figure 2).

17. Warm PBS, TrypLE, and cell media in 37°C water bath
18. For adherent cells, wash cells twice with PBS and remove by aspiration.
19. Add TrypLE to virally transduced cells (50 μ L per 12-well), incubate for 2 min at 37°C and 5% CO₂
 - a. Check for adequate detachment after 1 min, can tap dish to mechanically dislodge cells
 - b. When 80% cells have detached, halt trypsinization by adding at least 3 \times volume complete media
20. Spin cells down at 900 g at room temperature and resuspend to 75,000 cells/50 μ L (for Ibidi 6-well plate) complete media
 - a. Tip Ibidi plate and add 50 μ L into the elevated port, pipetting as close to the plate as possible, to ensure the cells spread between ports, incubate for 20 min at room temperature in the sterile hood.
 - b. Add 60 μ L complete media to each port to prevent drying
21. Incubate overnight (12–24 h) to ensure attachment

△ CRITICAL: When plating cells, make sure the solution of suspended cells is adequately mixed. This will ensure the sample number of cells makes it into each well and therefore the cells will be at the same density. We note changes in cell density may have an effect on their metabolism.

Prepare injection solutions: Day 3

⌚ Timing: 15 min

Pharmacological perturbation requires two solutions which must be made fresh before each experiment. Since Laconic measures glycolysis, it is important to block any mitochondrial consumption of pyruvate which is accomplished with rotenone. Nigericin is an ionophore used to clamp the intracellular pH.

22. Warm Fluorobrite media and ICB in 37°C water bath.
23. Thaw stock solution of pCMBA and nigericin on ice.
24. Mix calibration curve solutions via dilution series (Table 2; Example calibration curve, Figure 3).
 - a. Make 10 μ M nigericin and 2 μ M of rotenone in 10 mL of ICB
 - b. Use stock solution of lactate to generate 2 mL of 10 mM lactate
 - c. 10-fold Dilution series from 10 mM to 0.0001 mM (dilute 6 \times) with each solution at 1 mL final volume in ICB with rotenone and nigericin
25. Mix MCT blockade solution for LPR (lactate production rate) assay (Table 3)
 - a. 500 μ M pCMBA in Fluorobrite with glucose or ECB with glucose, 1 mL per well of cells
 - b. Bring 1 mL of Fluorobrite or ECB per well to wash cells prior to imaging

Note: Calibration curve only needs to be performed once. When performing LPR experiments, calibration curve can be omitted (step 24).

⏸ Pause point: Solutions can stay at room temperature for up to an hour.

Perform imaging of intracellular lactate or lactate production rate (LPR): Day 3

⌚ Timing: 2–3 h

Injection of solutions and data acquisition at the microscope. This section can be used with any data acquisition, including calibration curve of intracellular lactate and measurement of intracellular

Table 2. Solutions for calibration curve

Solution #	Component	Stock solution	Final concentration	Amount (μL)
1	ICB	1×	1×	988.6
	Rotenone	2.5 mM	1 μM	0.4
	Nigericin	10 mM	10 μM	1
	Lactate	1 M	10 mM	10
2	ICB	1×	1×	988.6
	Rotenone	2.5 mM	1 μM	0.4
	Nigericin	10 mM	10 μM	1
	Lactate	100 mM	1 mM	10
3	ICB	1×	1×	988.6
	Rotenone	2.5 mM	1 μM	0.4
	Nigericin	10 mM	10 μM	1
	Lactate	10 mM	100 μM	10
4	ICB	1×	1×	988.6
	Rotenone	2.5 mM	1 μM	0.4
	Nigericin	10 mM	10 μM	1
	Lactate	1 mM	10 μM	10
5	ICB	1×	1×	988.6
	Rotenone	2.5 mM	1 μM	0.4
	Nigericin	10 mM	10 μM	1
	Lactate	100 μM	1 mM	10
6	ICB	1×	1×	988.6
	Rotenone	2.5 mM	1 μM	0.4
	Nigericin	10 mM	10 μM	1
	Lactate	10 μM	0.1 mM	10

Solutions for the calibration curve should be made the same day as imaging. Intracellular buffer (ICB) will be used as the solvent with varying concentrations of lactate in the presence of nigericin and rotenone for permeabilization. Lactate should first be serially diluted 1:10 5× then used to make the following solutions.

glycolysis (lactate production rate, LPR). Calibration curve only needs to be performed once unless the optical set-up is changed (filter sets, for example).

26. Turn on light source, microscope, computer, imaging software (such as Micromanager), and additional controls such as stage, CO₂, and temperature. Wait at least 1 h for the microscope to thermally stabilize.
27. Serum starve the cells in the Ibidi plate prior to imaging
 - a. Warm imaging solution (Fluorobrite or ECB) to 37°C
 - b. Replace complete media in Ibidi plate containing cells with imaging solution
 - i. On the left port of the Ibidi plate, remove 90 μL of media. To the right port, add 90 μL of imaging solution. Perform 3 exchanges in total to completely remove the complete media.
 - c. Incubate for at least one hour prior to imaging at 37°C
28. Wash the tubing with at least 5× volume with a 10 mL syringe filled with PBS then assemble imaging chamber.
 - a. Connect input port before output port.
 - i. After the final rinse, inject imaging solution (Tables 2 and 3) into the input line using a needle and 1 mL syringe through the in-line injection port, holding the male elbow connector end at the same height as the injection port in order to ensure the entire tubing contains imaging solution liquid before adding the cap (Figure 2). Capping the input line to close the input port will prevent fluid from leaking out of the input line into the imaging chamber. However, some diffusion is inevitable. Therefore, perform imaging away from the input port.
 - ii. Fill the ports on the Ibidi plate with the imaging solution using a 200 μL pipette so that there is no air within the port.

Table 3. Example solution calculator for LPR injection solutions

Solution #	Component	Stock solution	Final concentration	Amount (μL)
1	ECB	1×	1×	990
	Glucose	1 M	10 mM	10
2	ECB	1×	1×	989
	Glucose	1 M	10 mM	10
	pCMBA	0.5 M	0.5 mM	1

Solutions should be prepared fresh on the same day as imaging. Extracellular buffer (ECB) or glucose-free Fluorobrite can be used as the solvent which should be warmed prior to solution preparation. If using glucose-containing Fluorobrite, glucose does not need to be added. The example calculations are dependent on 1 mL solution injections which should be adjusted depending on the length (and therefore volume) of the tubing.

- iii. Attach the input line first, press down firmly to ensure attachment. Then repeat attaching the output line.
 - b. Ensure waste line is securely placed into waste container and that the input line is accessible without disturbing the imaging focus.
29. Set imaging parameters
 - a. Exposure should be set to 100–200 milliseconds at a frame rate of 1 frame/2 s in both donor and FRET channels to minimize photobleaching.
 - b. Choose a representative field of view away from the injection port and edges of well or multiple fields of view if you have stage control. [Troubleshooting 1](#), [Troubleshooting 2](#).
30. Start imaging with injections
 - a. Attach a 25g needle to a new 1 mL syringe, used for injecting drugs
 - i. Pull liquid up into the syringe slowly to avoid bubble formation
 - ii. Flick syringe as needed to eliminate bubbles
 - iii. Use a different needle/syringe for each subsequent injection
 - b. Start imaging. Pause image acquisition if necessary to inject each subsequent solution. We perform this action slowly (1 mL over 5 s) to avoid cavitation and bubble formation. If doing the intracellular lactate calibration curve, perform those injections in increasing lactate concentration. Otherwise, follow the injection protocol for LPR. See [Tables 2](#) and [3](#) for defined solutions. [Troubleshooting 3](#), [Troubleshooting 4](#)
 - c. Acquire at least 2 min of data after each injection.
31. At the end of the assay, dispose of the cells unless using them for further analysis.
 - a. Wash the tubing 3× with 10× volume using 70% ethanol
32. Transfer data and turn off imaging equipment

△ **CRITICAL:** The complete media contains growth factors that can activate the cells, thus changing their metabolism. Therefore, cells must be starved for at least one and up to two hours prior to imaging.

△ **CRITICAL:** Assembly requires attention to eliminating air bubbles. Additionally, if you connect the waste line first, you may pull liquid from the plate out, leaving the cells dry and the plate susceptible to bubble formation. If a bubble is injected into the imaging chamber, cells could be ripped off the plate.

Note: To prevent bubble formation, place Ibidi chamber and tubing in incubator for 10 min prior to starting injections to bring everything to the same temperature. Make sure all solutions are at 37°C.

Note: Exposure time will be dependent on the expression level, light source strength, objective, and camera sensitivity. We suggest that the histogram of intensities fills at least $\frac{1}{4}$ the dynamic range of the camera. [Troubleshooting 1](#)

Note: After each injection, check focus to ensure that there has not been any focus drift. [Troubleshooting 4](#)

▮▮ **Pause point:** Data can be analyzed at any time.

Construct deep learning network for semantic segmentation

⌚ **Timing:** 3–7 days, depending on size of data set, computer memory and speed

For successful discrimination of metabolic subpopulations, accurate cell segmentation is critical. We therefore employed semantic segmentation for endothelial cells which have complex morphology. One increasingly powerful method for rapid segmentation is the use of artificial intelligence which has achieved human-level performance ([Ronneberger et al., 2015](#)). A subset of AI is deep learning, in which convolutional neural networks are used to perform recognition tasks, where many of the rules governing object assignment are hidden, as opposed to rule-based segmentation where all rules must be a priori explicitly laid out. For accurate Deep Learning to perform semantic segmentation, having a large ground truth dataset for network training is essential. A sample ground truth dataset can be downloaded at ([Harrison, Devin et al., 2021](#)), ground_truth_v2.zip. Code for this section can be downloaded and followed along at: https://github.com/wulab-code/STAR_methods. Please open “code_testing_new.m” for sample code. Be sure all scripts are in the same directory to eliminate path issues.

33. Generate ground truth dataset: Try to make conditions of ground truth dataset as close to how data will be taken so that the AI can do the least amount of guessing. Transfect with laconic as above on a glass substrate such as Ibidi 8-well or Labtek 8 well to facilitate high resolution microscopy. The goal here is to generate microscopy datasets that will be easily labeled as cell body (using transfected Laconic), cell boundaries (using VE-CADH antibody), nuclei (with Hoechst dye), and background.
 - a. Fix cells: After cells have grown to confluency and are appropriately expressing Laconic, wash cells with PBS at 37°C once and immediately fix using 4% PFA for 10 min at room temperature.
 - b. After fixation, wash with PBS 3×, 5 min each, on a rocking platform at room temperature.
 - c. Wash 3× with PBS, 5 min each, shaking, room temperature.
 - d. Block with blocking buffer for 30 min at room temperature, shaking
 - e. Primary antibody overnight (12–24 h) 1:100 VE-CADH in blocking buffer on a rocking platform at 4°C.
 - f. Wash with PBS 3×, 5 min each, shaking, room temperature.
 - g. Secondary antibody: 1:1000 goat anti-rabbit 549 for 1 h in blocking buffer at room temperature, shaking
 - h. Wash with PBS 3×, 5 min each, shaking, room temperature.
 - i. Stain nucleus: treat samples with 1:1000 Hoechst dye in PBS for 10 min at room temperature, then rinse with PBS once.

▮▮ **Pause point:** After cells have been fixed, they can be stored at 4°C in PBS for imaging later. If the chambers are properly sealed to prevent drying out (plastic wrap or parafilm) then cells can be imaged months later. Mounting media can be added as a preservative and cells can be stored at 4°C for years.

- j. Take images at same magnification / settings as experiment, as many as possible – green channel which will see laconic and label cytoplasm, red channel which will image VE-CADH and the cell boundary and blue channel which will image Hoechst and hence the nucleus of the cell. [Troubleshooting 5](#)

34. Label ground truth images: After generation of ground truth data set ([Harrison, Devin et al., 2021](#)) now we have to label all the pixels in the ground truth images. There are many ways to do this, including by manual inspection, or with custom image segmentation software. One highly accessible approach is to use free, open-source software such as CellProfiler ([Carpenter et al., 2006](#)). CellProfiler can use the nuclei as seeds for the cells then use the gradient of Laconic towards the cell edge as a marker for the boundary of the cell.
- Download CellProfiler ([key resources table](#)) and load images according to your folder and file structure
 - Calculate the illumination function for your microscope. Use “CorrectIlluminationCalculate” function. Select the input image and name the output image. Calculate the Background image and do not dilate the objects in the final averaged image. Select a block size which is larger than the object (a cell); in this case, 150 is sufficient. Do not rescale the illumination function. Calculate the function based for each image. Smooth the background image with a Gaussian function.
 - Perform background correction on all images. Use the “CorrectIlluminationApply” function to perform background correction. Select the input images (these are your original images) and name the output image. Use the illumination function calculated in step 34b. The illumination function should be subtracted from the input image.
 - Identify primary objects (nuclei). Use the “IdentifyPrimaryObjects” function. Select input image for Hoechst based images. Enter the typical diameter of the object in pixel units. Discard objects outside the diameter range; discard objects touching the border of the image. Use an adaptive thresholding strategy with Otsu thresholding method, with three-class thresholding. Assign pixels in the middle intensity range to the foreground. Threshold smoothing scale and correction factor can be modified but default settings work. Choose adaptive window of 50 and distinguish clumped objects by shape. Use propagation method to draw dividing lines between clumped objects. Automatically calculate the size of the smoothing filter for declumping, automatically calculate the minimum allowed distance between local maxima, and speed up by using a lower-resolution image to find local maxima. Fill holes in identified objects after both thresholding and declumping. Typically, this will result in detection of nuclei that are bigger than what is perceived by the user.
 - Shrink the nuclei. Use the “ExpandOrShrinkObjects” function. Select the input object for nuclei and shrink the objects by a specified number of pixels. This has to be eyeballed, but 3 is sufficient.
 - Now identify the cell bodies. Use “IdentifySecondaryObjects.” Select input image for Laconic based images, and select the shrunken nuclei as the input objects. Select propagation as the method to be used to identify secondary objects. Use an adaptive thresholding strategy with Otsu thresholding method, with three-class thresholding. Assign pixels in the middle intensity range to the foreground. Threshold smoothing scale and correction factor can be modified but default settings work. Choose adaptive window of 50 and regulation factor of 0.05. Fill holes in identified objects and do not discard objects touching the border of the image.
 - Now identify cytoplasm. Use “IdentifyTertiaryObjects.” Select the cell body objects from step 34f. Select the smaller identified objects (shrunken nuclei). And do not shrink smaller objects prior to subtraction.
 - Fill objects. Now that the cytoplasm has been identified, use “FillObjects”. This function will fill in any unmarked cytoplasm, using a minimum hole size of 54 and planewise fill.
 - To create cell boundaries, shrink the cytoplasm by 1 using the “ExpandOrShrinkObjects” function.
 - Shrink the nuclei by 1 pixel using the “ExpandOrShrinkObjects” function.
 - Now using the shrunken cytoplasm, identify the cytoplasm object by using “IdentifyTertiaryObjects.”
 - Now using the shrunken nuclei, identify the nuclei object by using “IdentifyTertiaryObjects.”

- m. Now using the cytoplasm with nuclei, identify the cytoplasm object by using "IdentifyTertiaryObjects" and smaller object is new shrunken nuclei.
 - n. Convert the following objects to image using "ConvertObjectsToImage" function: Filled cytoplasm, cytoplasm with nuclei, cytoplasm outlier, shrunken nuclei, shrunken nuclei_1 and shrunken cytoplasm
 - o. Save images to disk as Portable Network Graphics (png) using "SaveImages" function. The rest of the protocol assumes that images are saved in png format.
35. Construct deep learning algorithm using the Deep Learning toolbox in MATLAB: MATLAB is advantageous in that minimal computer programming is required. Smallest feature size will determine how many convolutions you need to perform. The idea is to use deconvolutions to increase the receptive field for each neuron and then localize where the neurons fire using upconvolutions. Finally, an inverse weight function will be used to account for unequal distributions of the pixel classes (background pixels are usually overrepresented, compared to boundary, cytoplasm, and nuclei, and could therefore bias the network). The first part will involve creating a datastore and using augmentation to increase the effective size of the dataset. The second part involves creating the segmentation network and assessing its performance.
- a. Download and install latest version of MATLAB with Deep Learning toolbox. Download the scripts from GitHub ([Link to code](#)). Open sample code "code_testing_new.m" in MATLAB.
 - i. Create a new script. All code referenced below are in "code_testing_new.m". The code is either directly shown below or referenced by line number. For each new line in the script, there will be a double arrow ">>" and can be typed directly into the script
 - b. This script assumes you are on a Windows PC. In this example, the "basedir" is the location of the groundtruth images with are .png format files. This script also assumes that you are using MATLAB 2018b although the script should be compatible with newer versions.

```
>> clear
>> addpath('C:\Program
Files\MATLAB\R2018b\examples\images\main')
>> basedir = 'D:\images\';
>> ground_truth = 'D:\images\groundtruth';
```

- c. Define classes.

```
>> classNames = ["background", "border", "cell", "nuclei"];
```

- d. Randomize into training and validation datasets. Pick the fraction of images you wish to use for training (frac_training) from the total number of groundtruth images (num_images), which the computer is blinded to when performing validation and testing. The script automatically divides the remaining non-training images into validation and testing evenly. A good starting point is 70% of images devoted to training. Num_images defines the number of groundtruth images.

```
>> num_images = 1000;
>> frac_training = 0.7;
>> [training, validation, testing] = ...
randomize_images(num_images, frac_training);
```

- e. Make and store training, validation, and testing datasets into different directories: one for the labeled images and one for the images themselves

```
>> divide_images(...  
basedir, ground_truth, training, validation, testing);
```

- f. Create image and pixel datastore for training and validation.

```
>> imds = imageDatastore(fullfile(basedir, 'training'),  
... 'FileExtensions', {' .png'});  
>> classNames =  
["background", "border", "cell", "nuclei"];  
>> pixelLabelIds = [0 1 2 3];  
>> pxds = pixelLabelDatastore(fullfile(basedir, ...  
'training_label'), classNames, pixelLabelIds);  
>> imds_validation = imageDatastore(fullfile(basedir, ...  
'validation'), 'FileExtensions', {' .png'});  
>> pxds_validation = pixelLabelDatastore(fullfile(...  
basedir, 'validation_label'), classNames, pixelLabelId  
s);  
>> pximds_validation = pixelLabelImageDatastore(...  
imds_validation, pxds_validation);
```

- g. Augmentation can be used to increase the amount of data available to the network. Define augmentation function.

```
>> augmenter = imageDataAugmenter('RandRotation', [-180  
180], ...  
'RandXReflection', true, ...  
'RandYReflection', true, ...  
'RandXShear', [0 10], ...  
'RandYShear', [0 10]);
```

- h. Define patch extraction datastore

```
>> patchds = randomPatchExtractionDatastore(...  
imds, pxds, 128, 'PatchesPerImage', 256, ...  
'DataAugmentation', augmenter);
```

```
>> patchds_validation =
    randomPatchExtractionDatastore(...
    imds_validation, pxds_validation, 128, 'PatchesPerImage', 256);
```

- i. Preview the augmented patches

```
>> minibatch = preview(patchds);
>> montage(minibatch.InputImage, 'DisplayRange', [0 255])
```

- j. Create inverse weighting function.

```
>> tbl = countEachLabel(pxds);
>> totalNumberOfPixels = sum(tbl.PixelCount);
>> frequency = tbl.PixelCount / totalNumberOfPixels;
>> classWeights = 1./frequency;
```

- k. Generate graph of the network (Figure 4). “numFilters” determines the number of channels in the output of the convolutional layer. “filterSize” is the size of the local regions that are convoluted and should therefore be matched to the smallest feature which in our case is a few pixels. 3 is the smallest filter size.

Lines 43–174

- l. Set options for the training. These are trial and error, but below are some starting points.

```
>> initialLearningRate = 0.05;
>> maxEpochs = 50;
>> minibatchSize = 16;
>> l2reg = 0.0001;
>> options = trainingOptions('sgdm', ...
    'InitialLearnRate', initialLearningRate, ...
    'Momentum', 0.99, ...
    'L2Regularization', l2reg, ...
    'MaxEpochs', maxEpochs, ...
    'MiniBatchSize', minibatchSize, ...
    'VerboseFrequency', 20, ...
    'LearnRateSchedule', 'piecewise', ...
    'LearnRateDropFactor', 0.1, ...
    'LearnRateDropPeriod', 20, ...
    'Shuffle', 'every-epoch', ...
    'Plots', 'training-progress', ...
    'GradientThresholdMethod', 'l2norm', ...
```

```
'CheckpointPath', fullfile(basedir, 'checkpoint'), ...  
'GradientThreshold', 0.05, ...  
'Verbose', true, ...  
'ExecutionEnvironment', 'multi-gpu');
```

- m. Train algorithm and save the network (Figure 5). This can take anywhere from overnight (12 h) to 1 week depending on the speed of your computer. You should see the accuracy increase and the loss decrease as the training attempts to converge. Accuracy should be above 80% and loss should be below 1 to have acceptable results. [Troubleshooting 6](#)

```
>> [net, info] = trainNetwork(patchds, lgraph2, options);  
>> save network.mat net
```

△ CRITICAL: We suggest imaging at least 20,000 cells for deep learning but any additional images will help. Only 70% of images will be used for training. The remaining 30% will be split evenly for validation and testing of the deep learning network.

Alternatives: Although we did not specify in the CellProfiler section (27), it is possible to use the VE-CADH signal as the boundary and the Laconic and Hoechst signal as markers for the cell body and nucleus when labeling cells for ground truth.

Note: If the computer runs out of memory, try reducing the minibatch size. If the learning rate is not fast enough, adjust the momentum, LearnRateDropFactor and LearnRatePeriod. While these are good starting points, this is entirely trial and error. Typically, the accuracy should increase almost immediately in an upward trajectory from 0 to 60% within one 50 iterations with slow improvement afterwards. While the speed of calculation depends ultimately on the amount, quality of data, and computer specifications, a typical run on a fast computer with 3 GPUs (circa 2018) takes 24 h.

Alternatives: It may be possible to use transfer learning using a published network to avoid creating one's own deep learning network.

[Troubleshooting 7:](#) MATLAB code has difficulty executing

▮▮ **Pause point:** As this all takes place in a computer, resuming can happen at any time.

Semantic segmentation

⌚ **Timing:** [4 h - days], depending on computer memory and speed, and size of dataset

36. Pre-processing images after imaging Laconic. Pre-processing images is important to remove imaging artifacts such as vignetting or uneven illumination, in order to have the best performance (assuming that the ground truth dataset is ideal). The network will likely perform well on unoptimized images as well. One can in theory test this under quantification and statistical analysis below. It is possible to take experiment images and process them first in CellProfiler, but here we opt to use MATLAB so that everything is "one-click". The first step is to eliminate vignetting and uneven illumination, using a tophat filter. However, to specify filter properties, one needs to know the imaging properties of the microscope. Start a new script. Specify the

camera and magnification parameters including pixel size (in real space). Specify the approximate size of a cell. A good rule of thumb is to make the tophat filter about 3× the size of a cell (Crocker and Grier, 1996). Specify the expected input image size for the network and bit depth. Make sure that the image size and bit depth are the same as that which was used in creating the deep learning network above (here, the image size was 1200×1200 and the bit depth was 8 bit).

```
>> pixel_size = 11e-6;
>> magnification = 10;
>> cellsize = 11e-6;
>> tophatw = 3*round(cellsize/(pixel_size/magnification));
>> h = fspecial('disk', tophatw);
>> thresh = 0;
>> targetimsize = [1200 1200];
>> targetbitsize = 8;
```

- a. Resize image and use tophat to get rid of long wavelength intensity fluctuations where venusimage and tfpimage are the files of the Venus image and the mTFP image, respectfully, although in principle it doesn't matter.

```
>> venusimage = 'venus.tif';
>> tfpimage = 'tfp.tif';
>> [venus_orig tfp_orig thresh_venus thresh_tfp] = ...
    resizetophatim(venusimage, tfpimage, ...
    targetimsize, h, thresh, targetbitsize);
```

37. Perform semantic segmentation of images: It doesn't matter if you do it on the background corrected Venus image or mTFP image. Here it is done on the Venus image
 - a. Load the network:

```
>> segnet = load('network.mat');
```

- b. Segment the image. It is much faster with a GPU than using a CPU.

```
>> output = semanticseg(venus_orig, segnet.net, ... 'ExecutionEnvironment', 'gpu');
```

Post-processing of images

⌚ Timing: [2–16 h], depending on computer memory and speed, and size of dataset

38. Post-processing after semantic segmentation: The output is a categorical array (1–4). The segmentation needs to be cleaned up as artifacts can be introduced. This script filters out small cells and small nuclei and saves the output. Convert the output image to a labeled image (see Figure 6 for example of original image, after semantic segmentation, and after additional post processing). See how well the segmented image compares to the original. In this example, the post processed image ("Additional filtering") has interpreted some noise as cells. This can be improved by increasing the size and/or accuracy of the ground truth data set.



```
>> [filtered_image filtered_nuclei] =
    filter_network_output(output);
>> bw_filtered_image = bwlabel(filtered_image);
>> bw_filtered_nuclei = bwlabel(filtered_nuclei);
>> figure
>> output_im = double(output);
>> bw_im = double(bw_filtered_image);
>> venus_im = imread(venusimage);
>> J = imadjust(venus_im, stretchlim(venus_im), []);
>> [X2] = output_im/max(output_im(:));
>> [X3] = bw_im/max(bw_im(:));
>> map = jet;
>> subplot(1,3,1)
>> imshow(J)
>> title('Contrast adjusted Venus image')
>> subplot(1,3,2)
>> imshow(X2, 'Colormap', map)
>> title('Semantic segmentation')
>> subplot(1,3,3)
>> imshow(X3, 'Colormap', map)
>> title('Additional filtering')
```

39. Decompose the segmented image into individual cells: The images have been segmented, but each pixel in each image is just now a number – 1–4 indicates background, cell boundary, cytoplasm, and nuclei. The cells have to be individually identified, which is the purpose of this part. Using parallel processing significantly speeds up the time required.

```
>> expand = 3;
>> venusresize = imresize(imread(venusimage), targetimsize);
>> tfpresize = imresize(imread(tfpimage), targetimsize);
>> imdata = cell_decomposition(bw_filtered_image, ...
    bw_filtered_nuclei, venusresize, tfpresize, expand);
```

40. Above, everything was done on one single image. Now we will put it into a loop using parallel processing. Place all variables that don't change outside the loop. We recommended that you use GPUs to do semantic segmentation first, then use regular CPUs (in parallel) for the rest. Use the demo_images.zip located here ([Harrison, Devin et al., 2021](#)). Unzip them into a directory titled "demo_images".
- First loop the part that requires GPU
Lines 227–248.

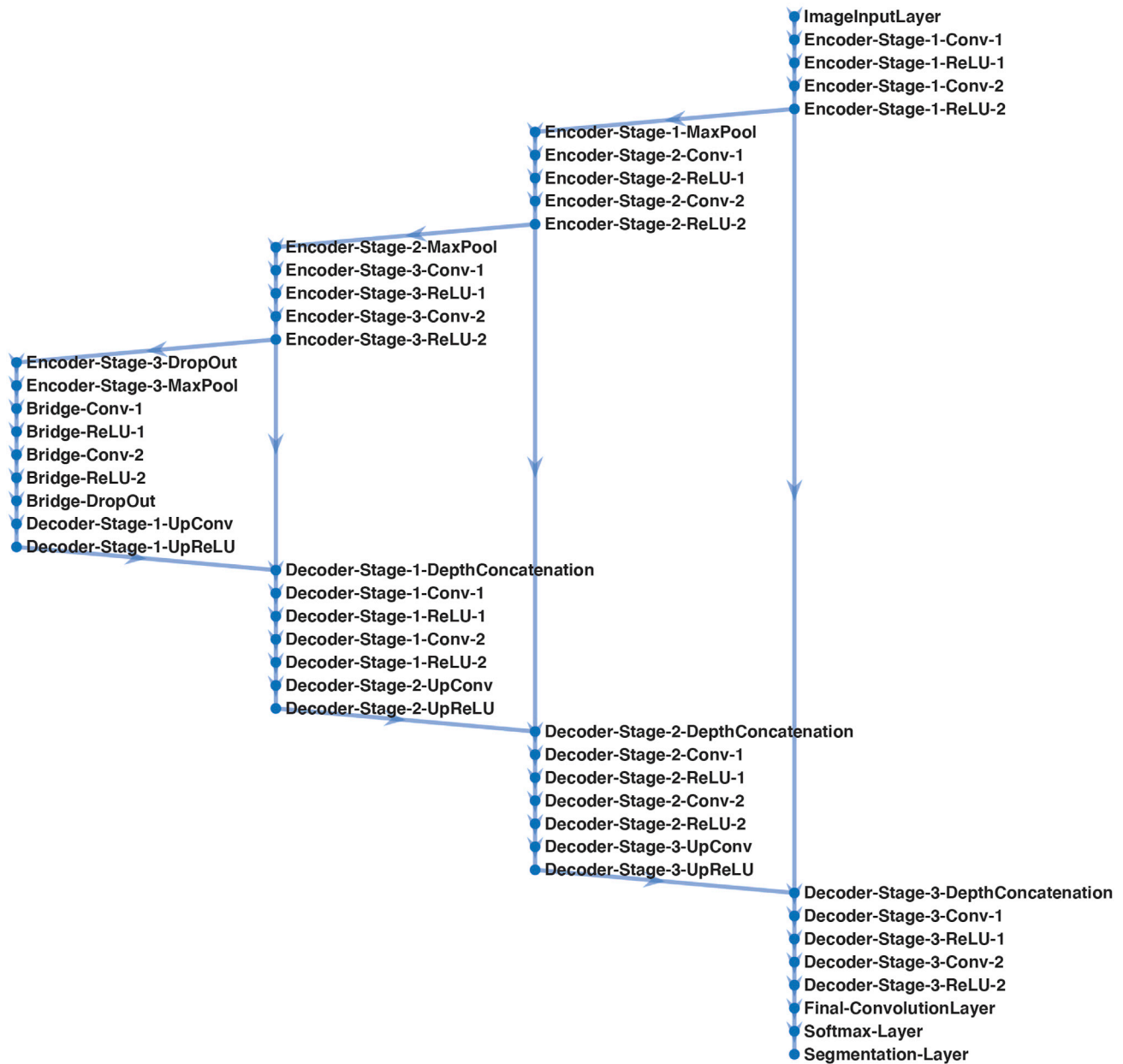


Figure 4. Deep learning architecture for semantic segmentation

Deep learning architecture for semantic segmentation, which consists of a contracting path (encoder) and an expansive path (decoder), forming a U-Net.

b. Next loop the part that requires CPU

Lines 250–278.

Note: A sample data set (in addition to the one referenced explicitly in the code above) is located at (Wu et al., 2021b) corresponding with figures in (Wu et al., 2021a). We recommend using the dataset corresponding to Figure 1e. Sample 1 is control and sample 2 is with DMOG. Injections for pCMBA occur at frame 120.

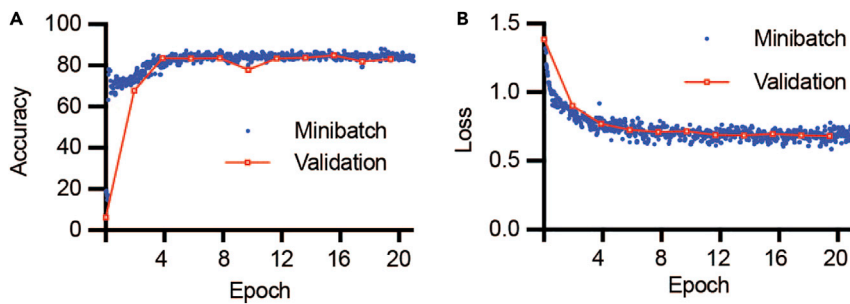


Figure 5. Training and validation of the convolutional neural network

Accuracy (A) and loss (B) as a function of training epoch. Training was stopped when there was no further change to loss or accuracy.

Link cell trajectories

⌚ Timing: 15 min

41. Linking together cell trajectories: Now that all images have been segmented, the next task is to link the intensities of each cell into a trajectory which is accomplished below by creating a matrix of cell positions and finding the nearest cell in an adjacent image. Save the output.

```
>> input_directory = 'singlecell_output';
>> [venus tfp venus_bg tfp_bg linkage] = ...
    cell_linking(input_directory, targetimsize);
>> output_directory = 'singlecell_links';
>> mkdir(output_directory);
>> save(fullfile(output_directory, 'linkvenus.mat'), 'venus');
>> save(fullfile(output_directory, 'linktfp.mat'), 'tfp');
>> save(fullfile(output_directory, 'linkvenusbg.mat'), 'venus_bg');
>> save(fullfile(output_directory, 'linktfpbg.mat'), 'tfp_bg');
>> save(fullfile(output_directory, 'linkage.mat'), 'linkage');
```

Analysis of data

⌚ Timing: 30 min

Now that the experiment is complete, we need to analyze the data and calculate the rate of intracellular glycolysis or the lactate production rate (LPR). To obtain the LPR, calculate the slope of the mTFP/Venus signal (the lactate signal) over a 40 s trajectory and fit it against a linear model during the portion of the time the cells were exposed to MCT1 inhibitor according to the following equation (Equation 1):

$$LPR = \frac{\Delta \text{Log}_{10} \text{Lactate}}{\Delta t}$$

42. Load the linked trajectories: At this point the segmented images in each channel (mTFP, Venus) have been linked through time in "linktfp.mat", "linkvenus.mat" which contains the mean

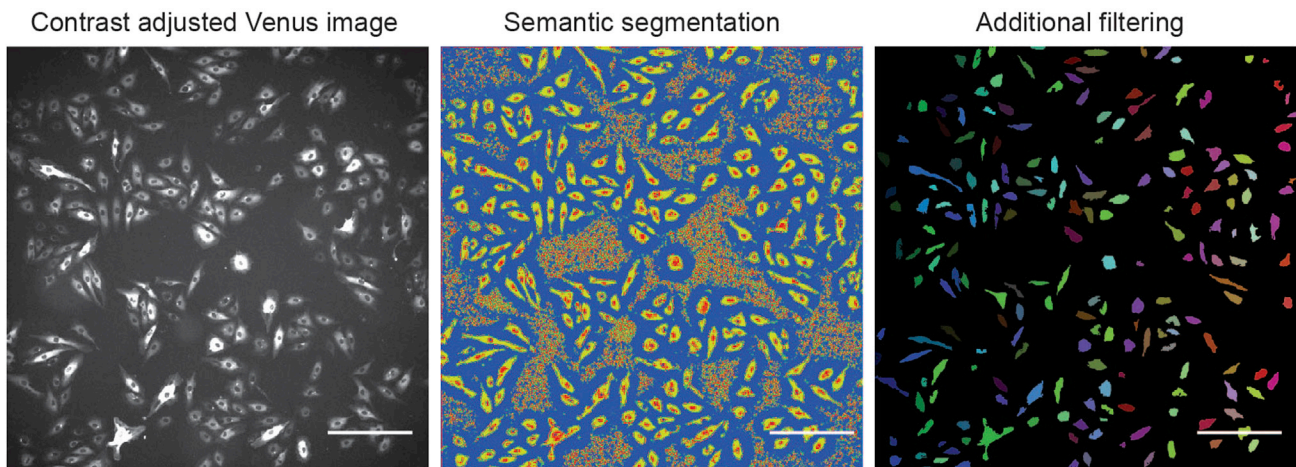


Figure 6. Single field-of-view segmentation example

The contrast adjusted original Venus image (left) undergoes semantic segmentation (center) which classifies cell boundaries but must be filtered additionally to remove non-cell objects (right). Scale bar is 264 μm .

intensity of the segmented cell in each respective channel. Background of the channels has been saved in "linktfpbg.mat" and "linkvenusbg.mat". These are loaded by selecting the appropriate directory and subdirectory containing the linked files:

```
>> load(fullfile(output_directory, 'linktfp.mat'));
>> load(fullfile(output_directory, 'linkvenus.mat'));
>> load(fullfile(output_directory, 'linktfpbg.mat'));
>> load(fullfile(output_directory, 'linkvenusbg.mat'));
```

43. Calculate the inverse FRET ratio. Inverse FRET ratio is calculated by dividing the mTFP by Venus, which were determined from the total cytoplasmic intensity. One can also check the background doesn't contribute significantly to the (inverse) FRET signal by calculating the FRET of the background. Each row of the "fret" or "fret_bg" matrix is the single cell inverse FRET ratio (Figure 7).

```
>> fret = tfp ./ venus;
>> fret_bg = (tfp - tfp_bg) ./ (venus - venus_bg);
```

44. Load the image data and set the LPR calculation over 40 s: Next, we calculate the LPR by fitting the rate of increase in mTFP/Venus signal over 40 s specified by dist (here, dist = 20 since we took 1 frame every 2 s).

```
>> imdata = load(fullfile('singlecell_output', ...
    [zerostr(5,0) '_data.mat']));
>> dist = 20;
```

45. Calculate the LPR for each solution change. Iterate through the fret variable, which is a matrix containing in each row the average fret value of a cell, and in each column the fret value of that cell as a function of time. The signal is filtered with a median filter of width 5. We look for changes in solution that tells us when to start calculating the slope. Keep on looking for changes in solution until you have run out ("solution_changes", specified a priori). Fit the relevant data with a line and calculate the slope. Save this variable. In this example (from the demo images dataset), solutions were injected in frame 150 (glucose) and frame 300 (glucose and pCMBA).

```
>> solution_changes = [150 300];
>> clear slope gof_r2 cellsize cells
>> for i = 1:size(fret,1)
    mfret = medfilt1(fret(i,:),5);
    for h = 1:length(solution_changes)
        if h ~= length(solution_changes)
            vec =
mfret(solution_changes(h):solution_changes(h+1));
        else
            vec = mfret(solution_changes(h):length(mfret));
        end
        idx = find(vec == min(vec));
        if h == length(solution_changes)
            if length(mfret) -
(solution_changes(h)+min(idx)) < dist
                idx = 1;
            end
        end
        x = solution_changes(h)+min(idx)-1;
        [fitobj gof] =
fit((x:x+dist)',mfret(x:x+dist)','poly1');
        slope(i,h) = fitobj.p1;
        gof_r2(i,h) = gof.rsquare;
    end
end
```

46. Now go through the fitted slopes and filter it for fit slopes that have an R^2 value that is high enough to ensure that the fit parameters correspond with the data (specified by "filter" parameter). In our hands, R^2 values > 0.8 are sufficient. Here, the slope has been fit into another variable. In the code below, we use R^2 value of 0.9 ('filter' variable).

```
>> filter = 0.9;
>> slope_n(1).slope = [];
>> for m = 1:length(slope(1,:))
    c = 1;
    for i = 1:size(slope,1)
        if gof_r2(i,m) > filter
            slope_n(m).slope(c) = slope(i,m);
            slope_n(m).r2(c) = gof_r2(i,m);
            slope_n(m).index(c) = i;
            c = c+1;
        end
    end
end
```

47. At this point, save the results in a file.

```
>> save results.mat slope_n
```

48. The LPR can be plotted either in MATLAB or another plotting program. The factor 0.01221 is the slope of the semilog change in lactate, as generated from the calibration curve (Figure 3), divided by the inverse frame rate (Equation 1). Sample data for Laconic calibration can be downloaded from (Wu et al., 2021b), Figure Extended Data 1a (Wu et al., 2021a). Injection times for different concentrations of lactate during calibration process are available here (Harrison, Devin et al., 2021).

```
>> glucose = slope_n(1).slope / 0.01221;
>> lactate = slope_n(2).slope / 0.01221;
```

49. Inspect a good trace (Figure 7 demonstrates a single cell trace of 1/FRET):

```
>> plot(fret(slope_n(2).index(50), 1:450), 'k.')
>> text(150, 0.6075, '\leftarrow Glucose injection')
>> text(300, 0.6075, '\leftarrow pCMBa injection')
>> xlabel('Frame')
>> ylabel('1/FRET')
```

Optional: Instead of calculating LPR (step 44), one can also directly compute the intracellular lactate concentration if there is a permeabilization step used as a reference (as for determining the inverse FRET/lactate calibration).

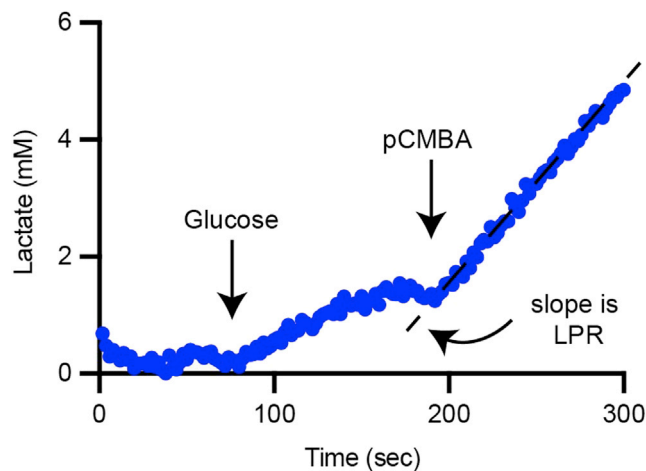


Figure 7. LPR single-cell trace

Example of the 1/FRET trace of an individual cell during the LPR assay. Arrows indicate solution changes.

Troubleshooting 8: Low number of useable FRET traces

EXPECTED OUTCOMES

This protocol allows the user to use deep learning to perform semantic segmentation of cells, dividing images of cells into nucleus, cytoplasm, cell boundary, and background. This data is all saved in .mat files on the hard drive and be easily used in other downstream applications or analysis. This deep learning strategy for segmentation can easily be applied to different cell types. This method of semantic segmentation is agnostic to Laconic and can be used in different cell labeling situations.

Generally, users should expect to generate FRET traces for hundreds of cells per field of view. Each individual trace is a time-series record of intracellular FRET. After the microscope system is calibrated, it is possible to map the FRET values to obtain absolute values of intracellular lactate. With knowledge of when the different compounds were injected, users can also calculate the lactate production rate, or glycolytic rate of individual cells. This protocol can be easily adapted to inspect the cell morphology, expression level of a given fluorescently labeled protein, or cell position and correlate it to LPR. This methodology can also be extended to use in different cell types. All the code is provided in GitHub, is open source, and free to be modified.

Here, we specifically demonstrate the LPR of a sample of HAECs treated with 500 μ m DMOG and with vehicle (DMSO) overnight (Wu et al., 2021a). DMOG stimulates HIF-1 α which increases glycolysis. LPR was calculated as above and exported to Prism (Figure 8).

QUANTIFICATION AND STATISTICAL ANALYSIS

Network performance estimation

There are many ways to estimate the error in machine learning or deep learning networks. Errors can be calculated for each class and a composite score can be calculated. Each estimation of error has its own positives and negative attributes, as error estimation is ultimately a little bit subjective, as one needs to choose which statistics to focus on. Common estimates of errors include the Matthew's Correlation Coefficient, Global accuracy, Class specific accuracy, Boundary F1 score, precision, recall and Dice score. Examples on how to calculate each below.

1. Save the network and run a test to see how accurate it is. It is important to only perform analysis on non-training data otherwise one will get artificially inflated scores. Calculate the global, class specific accuracy, and boundary F1 score.

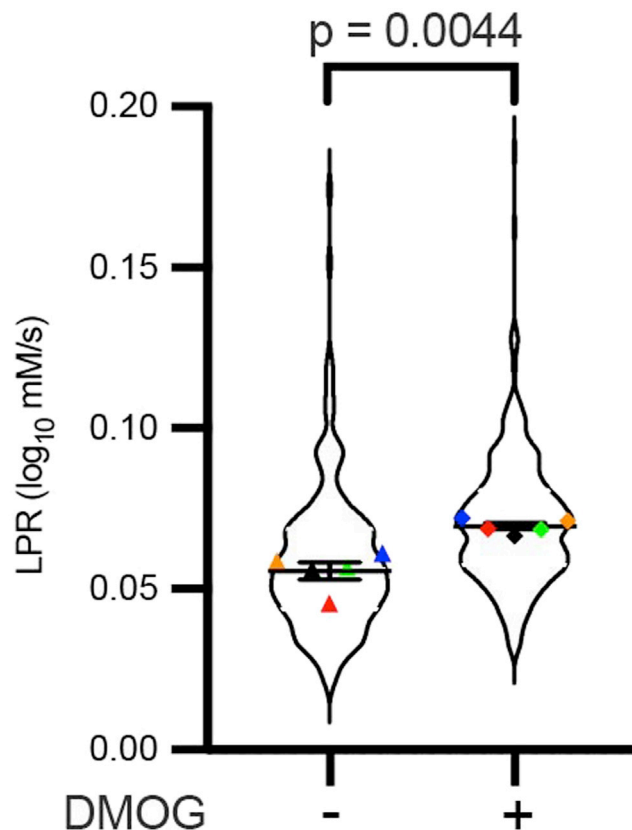


Figure 8. LPR bulk data for distinguishable populations in HAECs treated with control vs DMOG

LPR assay was performed on vehicle DMSO-treated (-) or DMOG-treated (+) endothelial cells (n = 126 -, 71 +; 164 -, 226 +; 181 -, 221 +; 120 -, 262 +; 191 -, 234 +; for 5 biological replicates error bars are SEM). p = 0.0044 by Student's t test.

Lines 358–370.

2. Calculate standard deviation.

Lines 380–414.

3. Calculate Matthews Correlation Coefficient where TP = true positive, TN = true negative, FP = false positive, FN = false negative.

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

Lines 372–377.

4. Calculate precision where TP = true positive, FP = false positive

$$Precision = \frac{TP}{TP + FP}$$

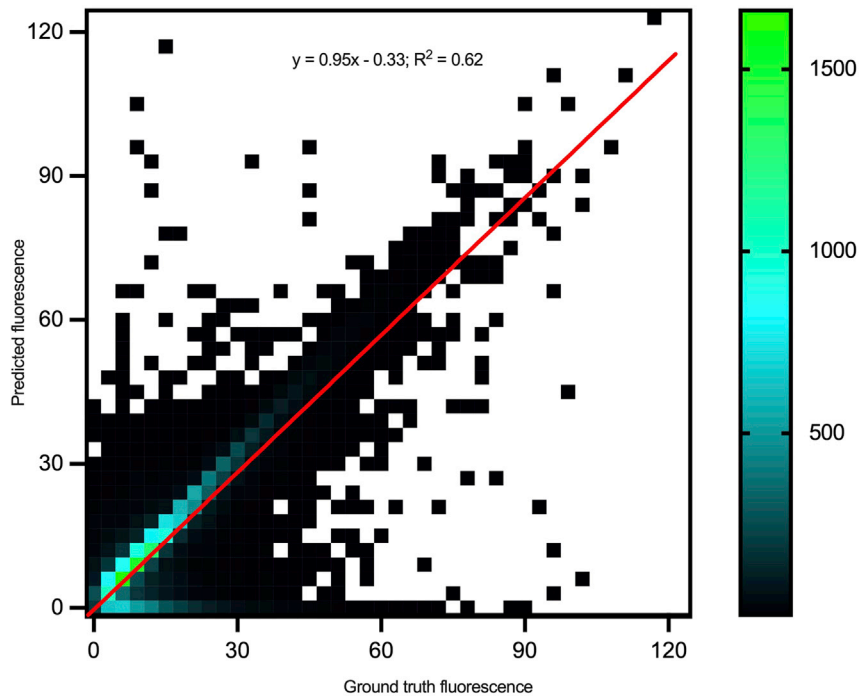


Figure 9. Ground truth fluorescence vs. predicted fluorescence

To evaluate the performance of segmentation, computed fluorescence is compared to ground truth fluorescence, defined as the signal inside the cell boundary. In this example, there is a consistent underestimation of the total cell fluorescence. The color code indicates number of cells. Pure black indicates 1 cell. White indicates 0 cells. The red line is the fit between ground truth and predicted fluorescence. Fit equation and goodness of fit is inset.

Lines 419–422.

5. Calculate recall where TP = true positive, FN = false negative.

$$\text{Recall} = \frac{TP}{TP + FN}$$

Lines 426–429.

6. Calculate the Dice score

Lines 434–449.

a. Calculate the mean Dice score for each class

Lines 451–454.

b. Calculate the standard deviation of the Dice score for each class

Lines 456–459.

Image segmentation error estimation

To estimate the error in deep learning assisted semantic segmentation, we first compute the error between ground truth fluorescence and segmented fluorescence. We use a test data set of cells expressing laconic for which we have ground truth and segmented these images with the deep learning algorithm. Then using the same image analysis pipeline, we plot the cell fluorescence as

estimated from deep learning networks against the ground truth cell fluorescence ($n \sim 20,000$ cells) (Figure 9). In this example, there is a consistent underestimation of the actual fluorescence after segmentation/deep learning. In part, this is because in the post-processing of segmented images cells are artificially shrunken then expanded during morphological operations. This parameter can easily be modified as needed (by modifying function 'filter_network_output.m'). This is also dependent on how "good" the deep learning network is – increase the size and quality of ground truth images such that the accuracy is $> 90\%$.

```

>> output_directory = 'network_test';
>> imds = imageDatastore(fullfile(basedir, 'testing'));
>> pxdsTruth = pixelLabelDatastore(fullfile(...
    basedir, 'testing_label'),classNames,pixelLabelIds);
>> pxdsResults = pixelLabelDatastore(output_directory, ...
    classNames,pixelLabelIds);
>> n_images = 65;
>> [cell_intensity_truth_total,cell_intensity_DL_total] =...
    evalTotalError(imds,pxdsTruth,pxdsResults,n_images,targetimsz);
>> save('error_estimation.mat', ...
    'cell_intensity_truth_total','cell_intensity_DL_total');
>> x = cell_intensity_truth_total';
>> y = cell_intensity_DL_total';
>> [fitobj gof] = fit(x,y,'poly1');
>> plot(cell_intensity_truth_total,cell_intensity_DL_total,'kx')
>> hold all
>> xax = linspace(min(cell_intensity_truth_total), ...
    max(cell_intensity_truth_total),100);
>> plot(xax,fitobj.p1*xax +fitobj.p2, 'r-','LineWidth',2)
>> xlabel('Ground truth fluorescence')
>> ylabel('Computed fluorescence')
>> text(60,120,['y = ' num2str(fitobj.p1) ' + 'num2str(fitobj.p2)])
>> hold off
  
```

Total LPR error estimation

To estimate the total estimated error in LPR, from semantic segmentation to LPR slope calculation, we turn to Monte Carlo simulations of LPR (bootstrapping). Using a dataset which provides a real measure between ground truth and fluorescence, we generate a line by randomly sampling ground truth fluorescence and the corresponding deep-learning generated fluorescence, 100 times. The slope of this line (ground truth vs. deep-learning) is then calculated by performing linear regression. This random sampling of fluorescence values, followed by slope calculation, simulates LPR calculation in real samples. The distribution of these bootstrapped slopes is plotted after repeating this 10,000 times, resulting in a Gaussian distribution of slopes (Figure 10).

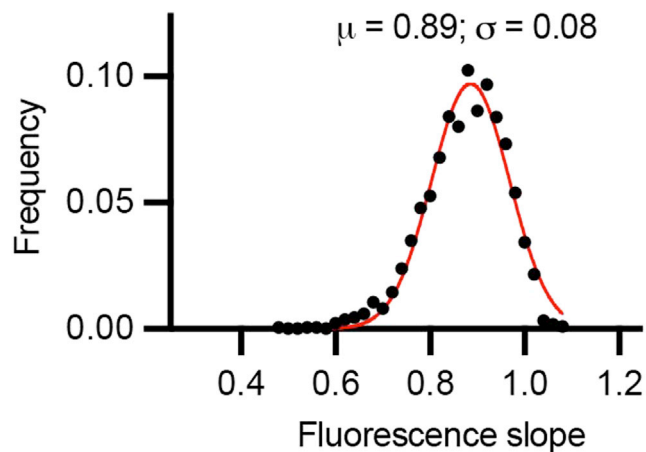


Figure 10. Total quantification of LPR

Data from [Figure 9](#) were randomly sampled 100 times and fitted to a line to get the slope, which represents a simulated LPR calculation. The black dots represent the frequency distribution of the simulated LPR after repeating this process 10,000 times. The red line is a Gaussian fit with mean 0.89 and standard deviation of 0.08. In this example, the deviation of the mean from 1.0 indicates that there is a consistent under-estimation of the “true” LPR, and a roughly 8% error. The deviation from 1.0 is likely due to the presence of consistent underestimation of total cell fluorescence, as in [Figure 9](#).

```
>> bootstrap_slope = [];
>> for i = 1:10000
    idx = randi([1 length(cell_intensity_truth_total)],1,100);
    x = cell_intensity_truth_total(idx);
    y = cell_intensity_DL_total(idx);
    fitobj = fit(x',y','poly1');
    bootstrap_slope(i) = fitobj.p1;
end
>> h = histogram(slope);
>> x = ((h.BinEdges(2:end) - h.BinEdges(1:end-1))/2) + ...
    h.BinEdges(1:end-1);
>> y = h.Values;
>> [curve fitobj gof] = fit(x',y','gauss1');
>> histogram(slope)
>> hold all
>> plot(curve)
>> text(0.2,750,['y = ' num2str(curve.a1) ...
    '*exp(-((x-' num2str(curve.b1) ')/' num2str(curve.c1)')^2)'])
```

Additional controls: Estimation of bleed through in microscope filter sets

Bleed through in FRET occurs when excitation of the acceptor occurs at the wavelengths used to excite the donor, leading to acceptor emission that does not occur due to FRET (acceptor bleed

through). Similarly, donor fluorescence can be detected in the acceptor channel (donor bleed through), artificially inflating FRET. A simple way to estimate degree of bleed through is to transfect cells with either mTFP or Venus without its pair and perform imaging as if imaging FRET. Then transfect cells with Laconic and image using the same parameters as those with mTFP and Venus alone. Compare the average signal level in minimal levels of lactate (maximal FRET) of each condition (mTFP alone, Venus alone, and Laconic).

To estimate the bleed through in the filter sets, create mRNA transcripts for mTFP and Venus separately (the FRET pair in Laconic), transfect the individual fluorophores as well as Laconic independently into HAECs and image.

7. Generate PCR templates for mTFP and Venus using laconic plasmid.
 - a. Generate template for *in vitro* transcription of mTFP via PCR using the T7-F and mTFPr primers (see [key resources table](#)) and protocol listed in [Table 1](#) and NEB Q5 Polymerase. The annealing temperature should be 60°C and the extension time is 25 s.
 - b. Generate template for *in vitro* transcription of Venus via PCR using the T7-Venus and BGH-R primers (see [key resources table](#)) and protocol listed in [Table 1](#) and NEB Q5 Polymerase. The annealing temperature should be 60°C and the extension time is 25 s.
 - c. Purify PCR products using QIAquick PCR Purification Kit ([Link to manufacturer's protocol](#))

Note: PCRs can be checked by running on a 1% agarose gel for the correct amplicon size (853 bp for mTFP and 890 bp for Venus).

8. Create mRNA of mTFP and Venus
 - a. Using the PCR-generated templates from 2c, create mRNA using mMACHINE™ T7 ULTRA Transcription Kit ([Link to manufacturer's protocol](#))
 - b. After cleaning and eluting mRNA, aliquot (100 ng/μL) and store at -80°C for minimal degradation. ([Link to manufacturer's protocol](#))
9. Transfect cells with mTFP, Venus, and Laconic as in mRNA transfection of Laconic above.
10. Image each fluorophore using the FRET filter sets after starving cells in glucose-free ECB for at least 1 h.
11. Compare donor channel and acceptor channel for all three transfections. The amount of donor bleed through (imaging mTFP transfection alone in the acceptor channel) and acceptor bleed through (imaging Venus transfection alone in the acceptor channel) can be subtracted from the donor and acceptor channels imaged using Laconic. We found, using the filter sets and illumination light source in this protocol, that bleed through was not greater than 5% of the laconic signal. Thus, it did not significantly affect the calculated FRET.

LIMITATIONS

Use of pCMBAs will be toxic to the cells so if cells are needed following the assay, Phloretin can be used instead but should be washed out immediately following imaging. Injections of bubbles can ruin experiments. Segmentation can only be optimized by acquiring the right annotated training data. Environmentally controlled imaging chamber is ideal but not completely necessary—this would require careful attention to pH of buffers.

TROUBLESHOOTING

Problem 1

Low expression level of Laconic

When selecting a field of view for performing imaging, you may notice low expression or heterogeneous expression such that each field of view minimizes the number of usable cells or data points. Additionally, the exposure time may have to be increased in order to boost your signal (step 29).

Potential solution

Dose titration of mRNA or virus

If expression levels are consistently low, we recommend dose titrating the mRNA or virus. If the expression levels are low for one or two experiments, consider changing aliquots. Both the laconic mRNA and adenovirus must be kept at -80°C for storage and freeze-thawed no more than twice. Adequate signal must be present to achieve accurate measures of intracellular lactate level. This can be tested by dose titration of mRNA or virus and performing intracellular lactate calibration (Figure 3). Some heterogeneity in cellular fluorescence is expected but, in our experience, does not affect intracellular lactate measurement as long as there is enough signal to noise.

Problem 2

Elongated cells

When selecting a field of view for performing imaging, you may notice altered morphology of the endothelial cells. We note this occurs more often when using mRNA transfection versus adenoviral transduction (see Figure 1) (step 29).

Potential solution

Adenoviral transduction

HAECs must not be used past passage 8. As the cells age, the morphology is more likely to change. We recommend using adenoviral transduction in lieu of mRNA if many experiments are to be performed for consistency of expression. Try avoiding using the same aliquot of defrosted cells for more than one month.

Problem 3

Injection causes cells to detach

Bubbles have been retained or formed in the imaging chamber or tubing (step 30).

Potential solution

Eliminate bubbles and make sure tubing and injection fluid is at the right temperature.

Bubbles may appear when the tubing is inserted into the Ibidi chambers. Flush the injection tubing slowly and thoroughly. Air may have been caught somewhere in the injection system. When attaching tubing to the chamber, make sure that no bubbles are entrained. When injecting, it must be done at a slow enough pace to not cause cavitation. Try reducing injection speed as well as to not dislodge any trapped air. Make sure all fluids are at the same temperature to prevent degassing.

Problem 4

Injection causes focus changes

After injecting solutions, focus is lost, likely due to rapid injection or accidental touching the microscope stage (step 30).

Potential solution

Equilibrate the microscope at the right temperature

First, fix the sample tightly on the stage to prevent accidental movement. Make sure the microscope has thermally equilibrated – turn on the scope and stage top incubator for at least an hour prior to imaging. Finally, re-focus the sample quickly after injection prior to acquiring data.

Problem 5

Too much intracellular labeling of VE-CADH

Occasionally, VE-CADH also stains the cell interior which will adversely affect cell boundary classification (step 33).

Potential solution

Try alternative microscopy approach or modify CellProfiler algorithm

If this is the case, either optimize staining with a different membrane marker, do not permeabilize, or try confocal microscopy and image different focal planes. In CellProfiler, one can also create a logical exclusion where overlap of VE-CADH and either Laconic or Hoechst causes VE-CADH to be deleted.

Problem 6

Deep learning accuracy does not reach above 80%

Deep learning is unable to accurately semantically segment cells and properly classify pixels in an image (step 35).

Potential solution

Ground truth is not accurate or there is not enough ground truth data.

Acquire more ground truth data. Carefully inspect the fixed samples and the output of CellProfiler and check to make sure that classification is accurate.

Problem 7

MATLAB code has difficulty executing

Errors are returned at the command prompt (step 35) .

Potential solution

Make sure that the working directory is correct.

Make sure that the working directory is the same directory as with all the scripts. Check through the code to ensure that the directory referencing the location of images is correct.

Problem 8

Low number of useable FRET traces

After processing imaging data with what appears to be hundreds of cells and adequate signal, only a few traces (out of hundreds) have the requisite R^2 for statistical testing (step 46).

Potential solution

Ensure the ECB is at the correct pH

The pH of the ECB could change after sitting in the 37°C, 5% CO₂ incubator for too long, since it is a bicarbonate-based buffering system. Store ECB in an open container in a 37°C, 5% CO₂ incubator for an hour in a small volume then measure pH. If the pH has changed substantially, make sure that the experiment is performed before this occurred. Alternatively, use Fluorobrite DMEM.

RESOURCE AVAILABILITY

Lead contact

Further information and requests for resources and reagents should be directed to and will be fulfilled by the lead contact, Yun Fang, yfang1@medicine.bsd.uchicago.edu.

Materials availability

This study did not generate new unique reagents.

Data and code availability

The protocol includes all datasets generated or analyzed for this study. Data and code are available without restriction. Data is available on Zenodo (<https://doi.org/10.5281/zenodo.4638059> and <https://doi.org/10.5281/zenodo.4898134>). All the code is provided in GitHub (https://github.com/wulab-code/STAR_methods).

ACKNOWLEDGMENTS

This work was supported by NIH grants R00AI106941 (J.H.), R21AI120010 (J.H.), R01HL138223 (Y.F.), and R01HL136765 (Y.F.), NIH New Innovator Award DP2AI144245 (J.H.), T32EB009412 (D.H.), T32HL007381 (D.H.), F32HL134288 (D.W.), NIH Pathway to Independence Award K99HL145113536 (D.W.), and R00HL145113 (D.W.). NVIDIA GPU Grant (D.W.), CSCTR Early Career Development Award (D.W.), and NSF Career Award 1653782 (J.H.) also supported this work.

AUTHOR CONTRIBUTIONS

D.H and D.W. designed and evaluated assay and wrote manuscript. Y.F. and J.H. wrote manuscript.

DECLARATION OF INTERESTS

The authors declare no competing interests.

REFERENCES

- Carpenter, A.E., Jones, T.R., Lamprecht, M.R., Clarke, C., Kang, I., Friman, O., Guertin, D.A., Chang, J., Lindquist, R.A., Moffat, J., et al. (2006). CellProfiler: image analysis software for identifying and quantifying cell phenotypes. *Genome Biol.* 7, R100. <https://doi.org/10.1186/gb-2006-7-10-r100>.
- Crocker, J.C., and Grier, D.G. (1996). Methods of digital video microscopy for colloidal studies. *J. Colloid Interface Sci.* 179, 298–310. <https://doi.org/10.1006/jcis.1996.0217>.
- Danthinne, X., and Imperiale, M.J. (2000). Production of first generation adenovirus vectors: a review. *Gene Ther.* 7, 1707–1714. <https://doi.org/10.1038/sj.gt.3301301>.
- Harrison, D., Wu, D., Fang, Y., and Huang, J. (2021). Human aortic endothelial cell ground truth dataset version 3 (Zenodo). <https://doi.org/10.5281/zenodo.4898134>.
- Ronneberger, O., Fischer, P., and Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, N. Navab, J. Hornegger, W.M. Wells, and A.F. Frangi, eds. (Springer International Publishing), pp. 234–241. https://doi.org/10.1007/978-3-319-24574-4_28.
- San Martin, A., Ceballo, S., Ruminot, I., Lerchundi, R., Frommer, W.B., and Barros, L.F. (2013). A genetically encoded FRET lactate sensor and its use to detect the warburg effect in single cancer cells. *PLoS One* 8, e57712. <https://doi.org/10.1371/journal.pone.0057712>.
- Wu, D., Harrison, D.L., Szasz, T., Yeh, C.-F., Shentu, T.-P., Meliton, A., Huang, R.-T., Zhou, Z., Mutlu, G.M., Huang, J., et al. (2021a). Single-cell metabolic imaging reveals a SLC2A3-dependent glycolytic burst in motile endothelial cells. *Nat. Metab.* 3, 714–727. <https://doi.org/10.1038/s42255-021-00390-y>.
- Wu, D., Harrison, D., Szasz, T., Yeh, C.F., Shentu, T.P., Meliton, A., Huang, R.T., Zhou, Z., Mutlu, G.M., Huang, J., et al. (2021b). Single-cell metabolic imaging reveals a SLC2A3-dependent glycolytic burst in motile endothelial cells (Zenodo). <https://doi.org/10.5281/zenodo.4638059>.