Cell Geometry

A web application for Cell Shape Analysis

Amil Khan, UCSB Electrical and Computer Engineering

Banff International Research Station

Mathematical Methods for Exploring and Analyzing Morphological Shapes across Biological Scales

GOALS OF THIS TALK

Introduce CellGeometry

Discuss 3D Cell Segmentation

Discuss Computing Cell and Nuclear Shape Modes

GOALS OF THIS TALK

Introduce CellGeometry

Discuss 3D Cell Segmentation

Discuss Computing Cell and Nuclear Shape Modes

And we are doing this

AT SCALE!


CELLGEOMETRY

Project Goal

The goal of this project was to build a web app that makes shape analysis techniques implemented in geomstats and similar projects accessible to non-technical users

Project Goal

The goal of this project was to build a web app that makes shape analysis techniques implemented in geomstats and similar projects accessible to non-technical users

```python
def exhaustive_align(curve, base_curve):
    """Align curve to base_curve to minimize the L² distance.

    Returns
    -------
    aligned_curve : discrete curve
    """
    nb_sampling = len(curve)
    distances = gs.zeros(nb_sampling)
    base_curve = gs.array(base_curve)
    for shift in range(nb_sampling):
        reparametrized = [curve[(i + shift) % nb_sampling] for i in range(nb_sampling)]
        aligned = PRESHAPE_SPACE.fiber_bundle.align(
            point=gs.array(reparametrized), base_point=base_curve
        )
        distances[shift] = PRESHAPE_SPACE.embedding_space.metric.norm(
            gs.array(aligned) - gs.array(base_curve)
        )
    shift_min = gs.argmin(distances)
    reparametrized_min = [
        curve[(i + shift_min) % nb_sampling] for i in range(nb_sampling)
    ]
    aligned_curve = PRESHAPE_SPACE.fiber_bundle.align(
        point=gs.array(reparametrized_min), base_point=base_curve
    )
    return aligned_curve


def preprocess(
    cells,
    labels_a,
    labels_b,
    n_cells,
    n_sampling_points,
    quotient=["scaling", "rotation"],
):
    """Preprocess a dataset of cells.

    if n_cells > 0:
        print(f"... Selecting only a random subset of {n_cells} / {len(cells)} cells.")
        indices = sorted(
            np.random.choice(gs.arange(0, len(cells), 1), size=n_cells, replace=False)
        )
        cells = [cells[idx] for idx in indices]
        labels_a = [labels_a[idx] for idx in indices]
        labels_b = [labels_b[idx] for idx in indices]

    if n_sampling_points > 0:
        print(
            "... Interpolating: "
            f"Cell boundaries have {n_sampling_points} samplings points."
        )
        interpolated_cells = gs.zeros((n_cells, n_sampling_points, 2))
        for i_cell, cell in enumerate(cells):
            interpolated_cells[i_cell] = _interpolate(cell, n_sampling_points)

        cells = interpolated_cells

    print("... Removing potential duplicate sampling points on cell boundaries.")
    for i_cell, cell in enumerate(cells):
        cells[i_cell] = _remove_consecutive_duplicates(cell)

    print("\n- Cells: quotienting translation.")
    cells = cells - gs.mean(cells, axis=-2)[..., None, :]

    cell_shapes = gs.zeros_like(cells)
    if "scaling" in quotient:
        print("- Cell shapes: quotienting scaling (length).")
        for i_cell, cell in enumerate(cells):
            cell_shapes[i_cell] = cell / basic.perimeter(cell)

    if "rotation" in quotient:
        print("- Cell shapes: quotienting rotation.")
        if "scaling" not in quotient:
            for i_cell, cell_shape in enumerate(cells):
                cell_shapes[i_cell] = _exhaustive_align(cell_shape, cells[0])
        else:
```

PAGE 1

Data Structure

- Load Data
  - Accepted Filetypes:
    - TXT/CSV
    - Zipped ROI Files from FIJI/ ImageJ
- Visualize Loaded Data
  - Interactive Visualization for Sanity Check

Cell 1

X	Y
548	-744
544	-740
544	-739
541	-736
540	-736
538	-734
536	-734
535	-733

Cell 2

613	-666
612	-667
610	-667
610	-668
609	-669
606	-669
605	-670
603	-670
602	-671

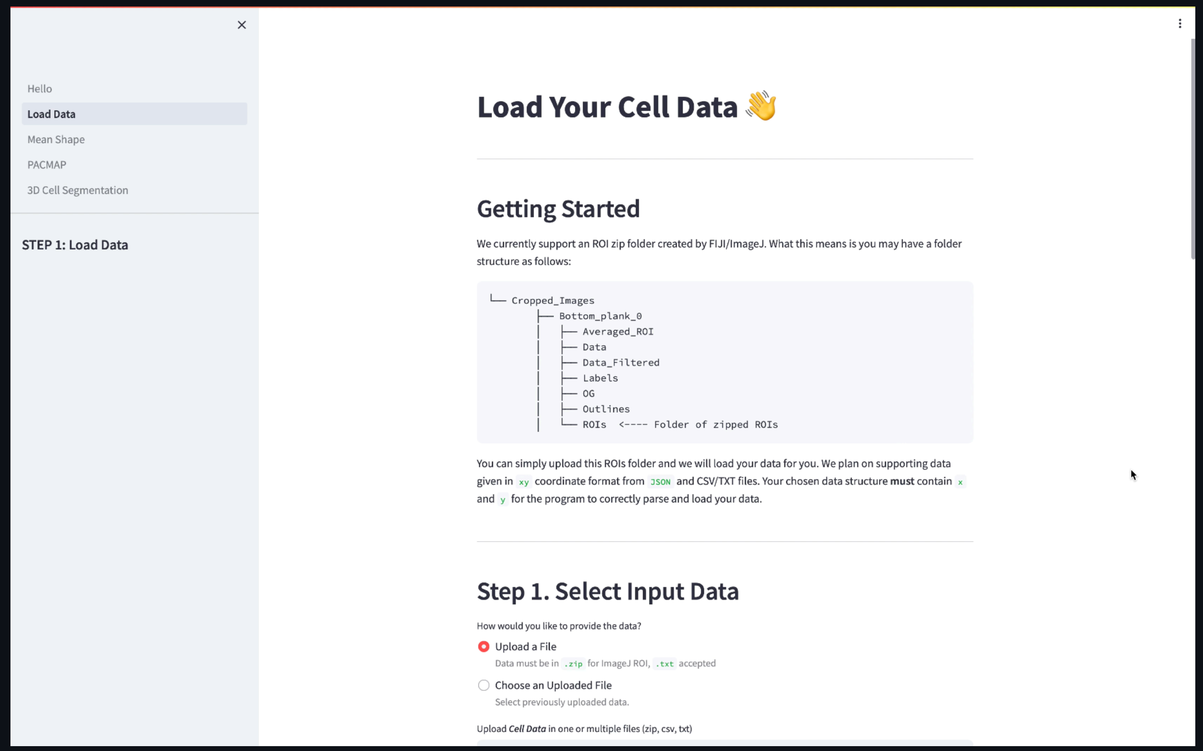
PAGE 1

Let’s Load some data!

- Load Data
  - Accepted Filetypes:
    - TXT/CSV
    - Zipped ROI Files from FIJI/ ImageJ
- Visualize Loaded Data
  - Interactive Visualization for Sanity Check

CELL GEOMETRY

Amil Khan, Samuel Feinstein, Adele Myers, Wanxin Li, Ashok Prasad, Khanh Dao Duc, Nina Miolane



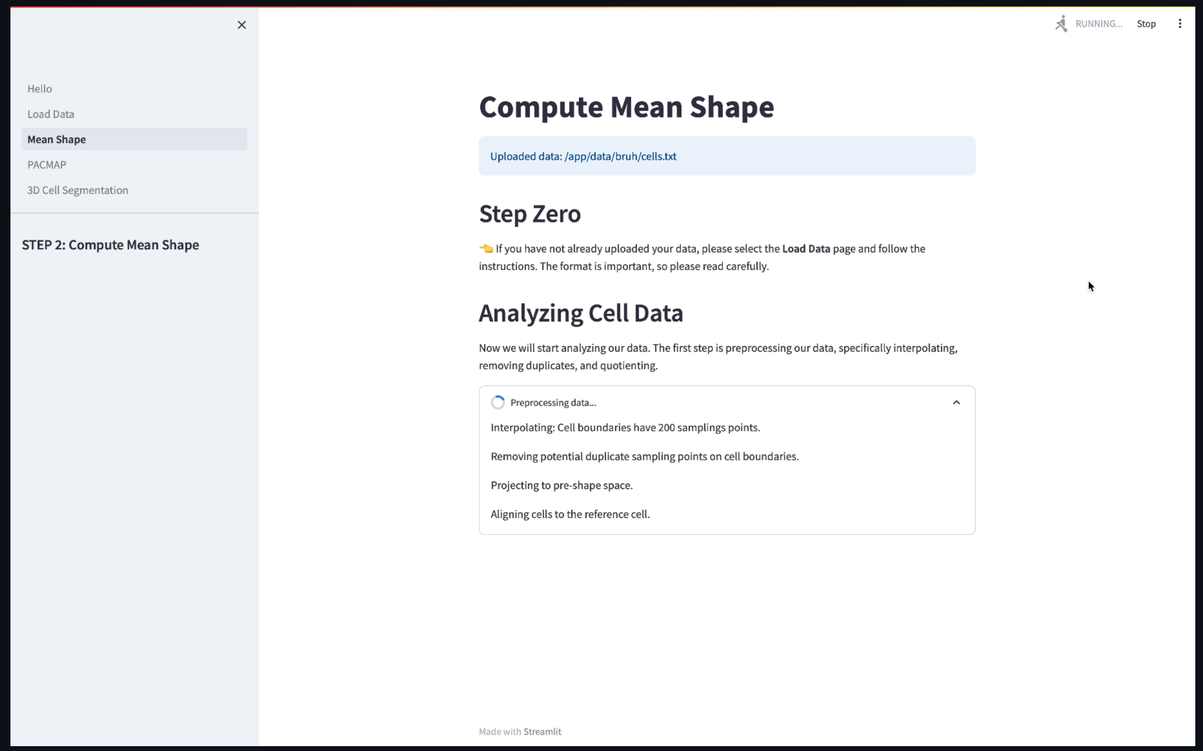
PAGE 2

Preprocessing Input Data

- Interpolation
  - Need discrete curves with the same number of sampled points to compute pairwise distances
- Remove Duplicates
  - During interpolation some of the discrete curves in the dataset are downsampled from higher number of discrete data points to lower number of data points

CELL GEOMETRY

Amil Khan, Samuel Feinstein, Adele Myers, Wanxin Li, Ashok Prasad, Khanh Dao Duc, Nina Miolane



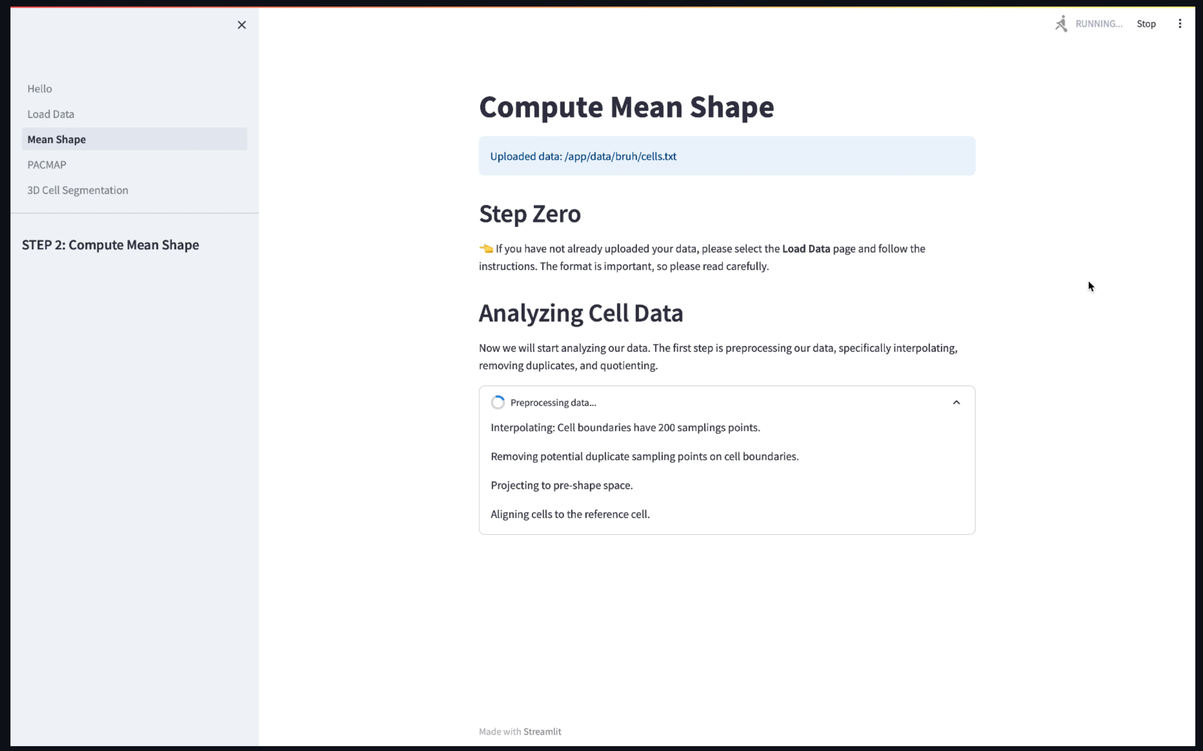
PAGE 2

Preprocessing

CELL GEOMETRY

Amil Khan, Samuel Feinstein, Adele Myers, Wanxin Li, Ashok Prasad, Khanh Dao Duc, Nina Miolane

- Projection to Pre-shape Space
  - We center (subtract the barycenter), rescale (divide by the Frobenius norm) and then align (find the rotation minimizing the L2 distance) two sets of landmarks.
  - These operations are performed by leveraging the geometry of the Kendall preshape spaces



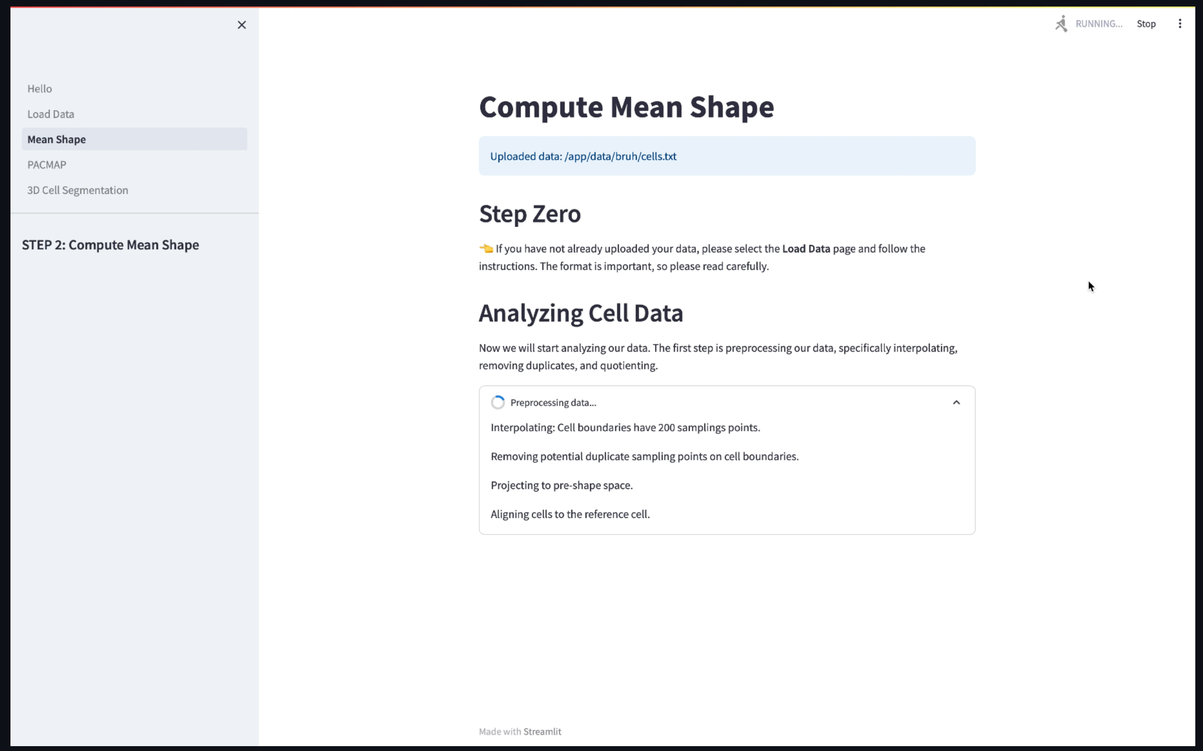
PAGE 2

Preprocessing

- Alignment
  - Since we are working with closed curves, the starting point associated with the parametrization of the discrete curves is also arbitrary.
  - We conduct an exhaustive search to find which parametrization produces the best alignment according to the above procedure (i.e. the distance to the base curve is the smallest)

CELL GEOMETRY

Amil Khan, Samuel Feinstein, Adele Myers, Wanxin Li, Ashok Prasad, Khanh Dao Duc, Nina Miolane



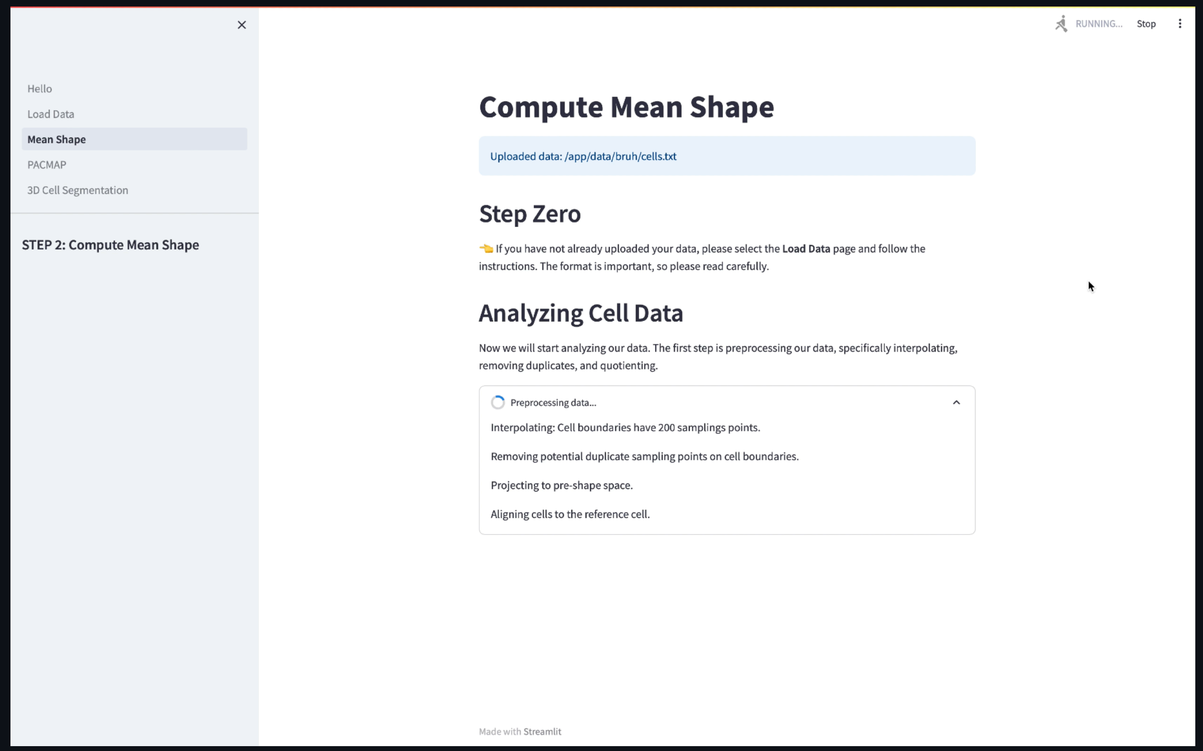
PAGE 2

Geodesic Trajectory

- Elastic Metric
  - Compute geodesics between discrete curves with respect to the elastic metric
  - These geodesics represent trajectories between cell boundaries that minimize an elastic energy, and the length of the geodesic defines a distance between curves

CELL GEOMETRY

Amil Khan, Samuel Feinstein, Adele Myers, Wanxin Li, Ashok Prasad, Khanh Dao Duc, Nina Miolane



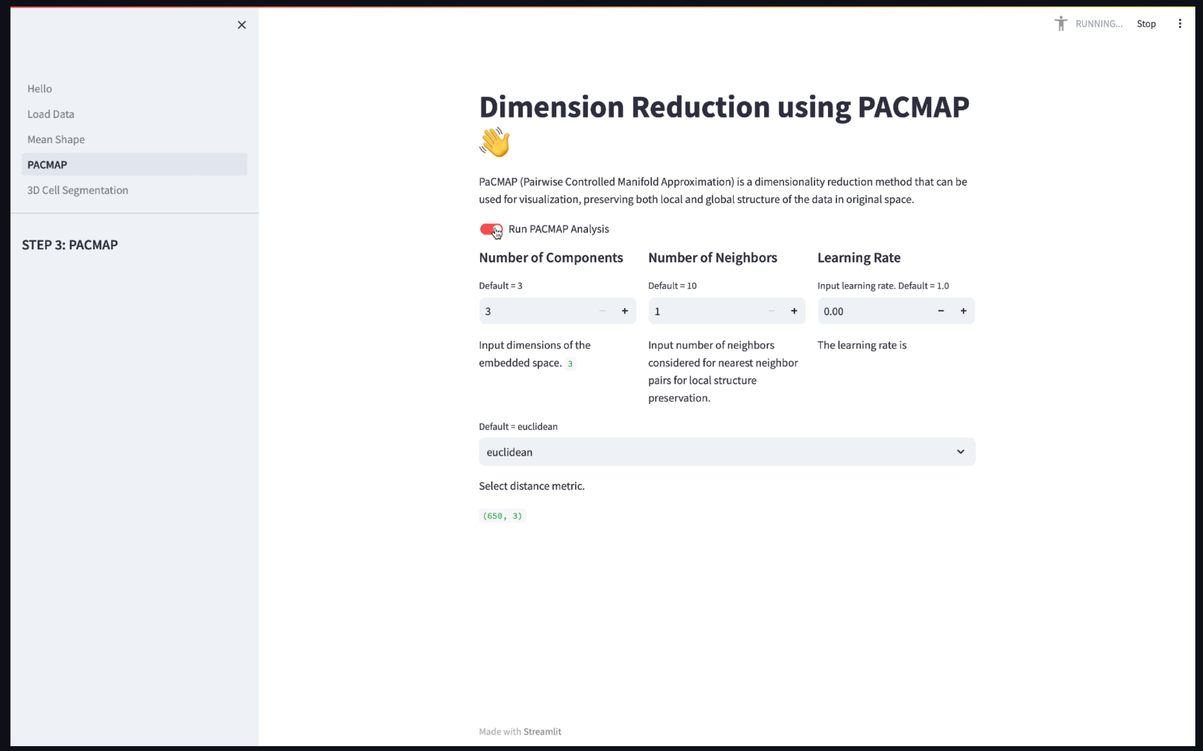
PAGE 3

Pairwise Controlled Manifold Approximation

- PACMAP
  - After computing the mean shape, click on **PACMAP** on the sidebar
- Visualization of PACMAP
  - We visualize the first 3 components, plot is automatically updated when params are changed

CELL GEOMETRY

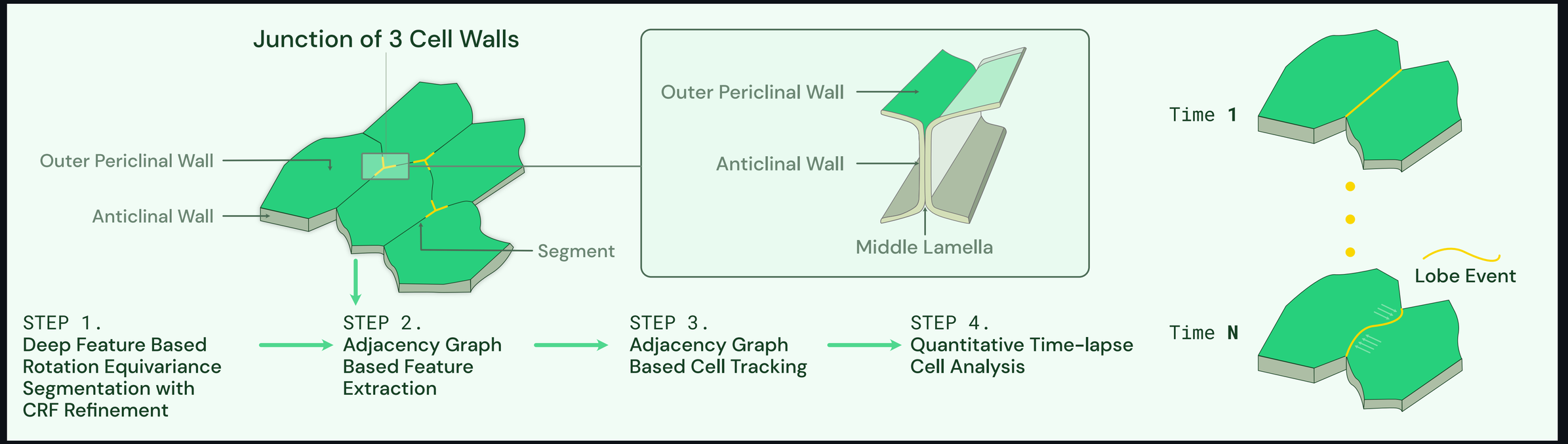
Amil Khan, Samuel Feinstein, Adele Myers, Wanxin Li, Ashok Prasad, Khanh Dao Duc, Nina Miolane



CELLECT 2.0

3D Cell Segmentation

Segmentation, tracking, and sub-cellular feature extraction in 3D time-lapse images

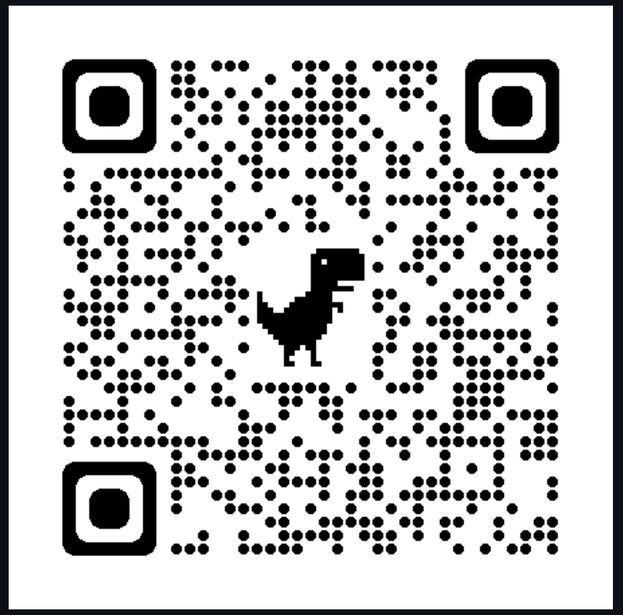


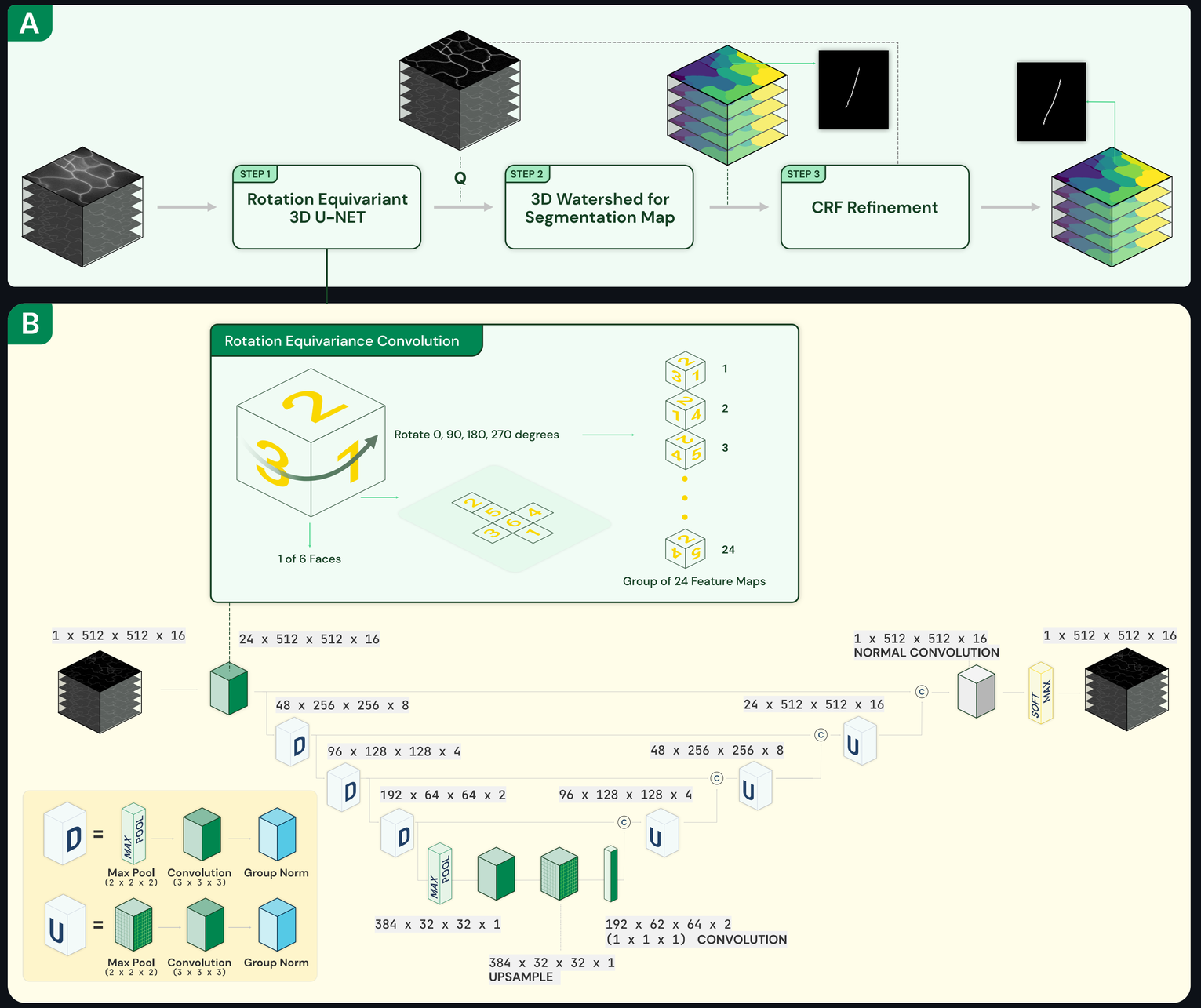
Jiang, J., Khan, A., Shailja, S. et al. Segmentation, tracking, and sub-cellular feature extraction in 3D time-lapse images. Sci Rep 13, 3483 (2023). https://doi.org/10.1038/s41598-023-29149-z

3D Cell Segmentation

Segmentation, tracking, and sub-cellular feature extraction in 3D time-lapse images

Read the Paper



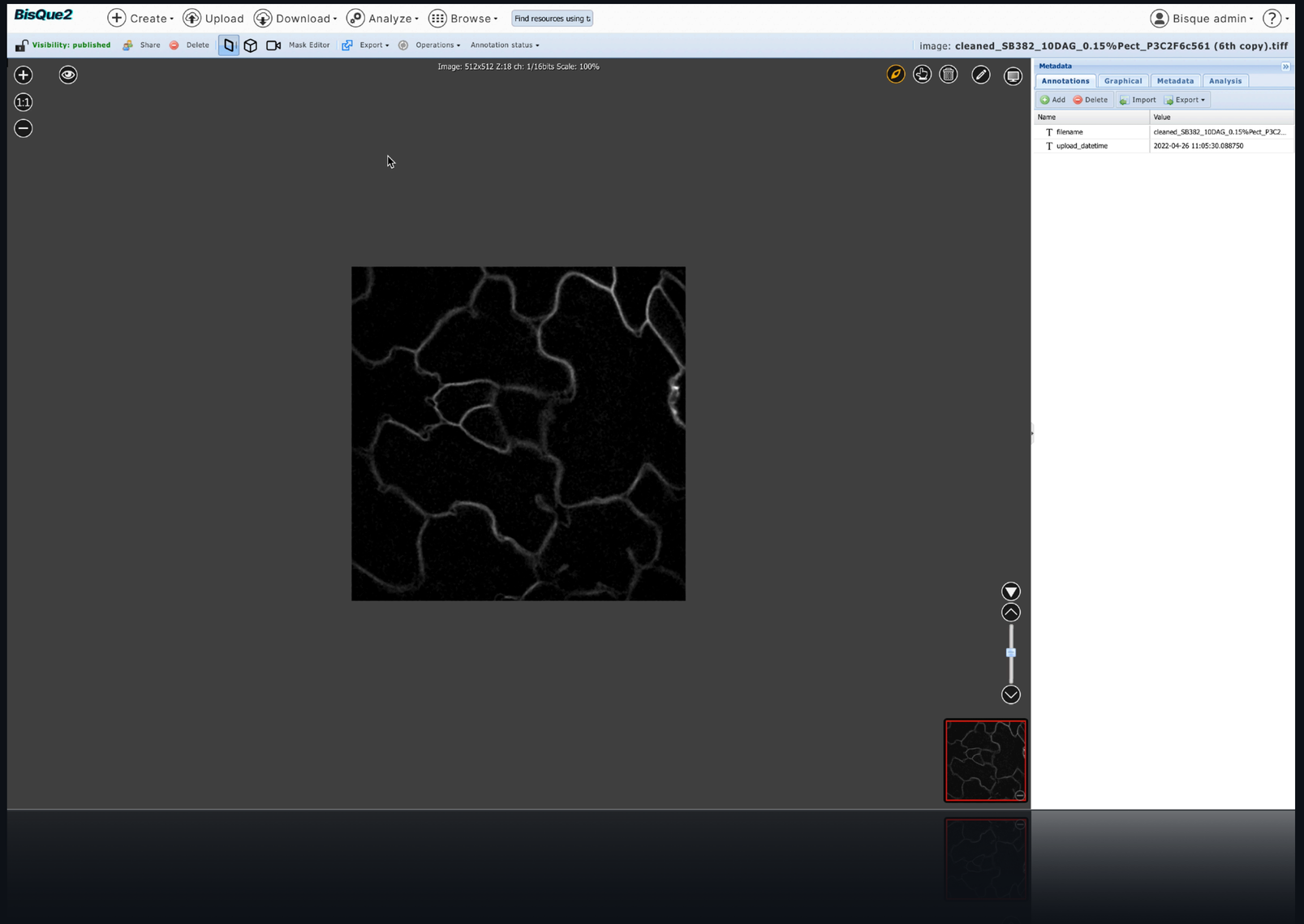


Jiang, J., Khan, A., Shailja, S. et al. Segmentation, tracking, and sub-cellular feature extraction in 3D time-lapse images. Sci Rep 13, 3483 (2023). https://doi.org/10.1038/s41598-023-29149-z

3D Cell Segmentation

Segmentation, tracking, and sub-cellular feature extraction in 3D time-lapse images

Read the Paper

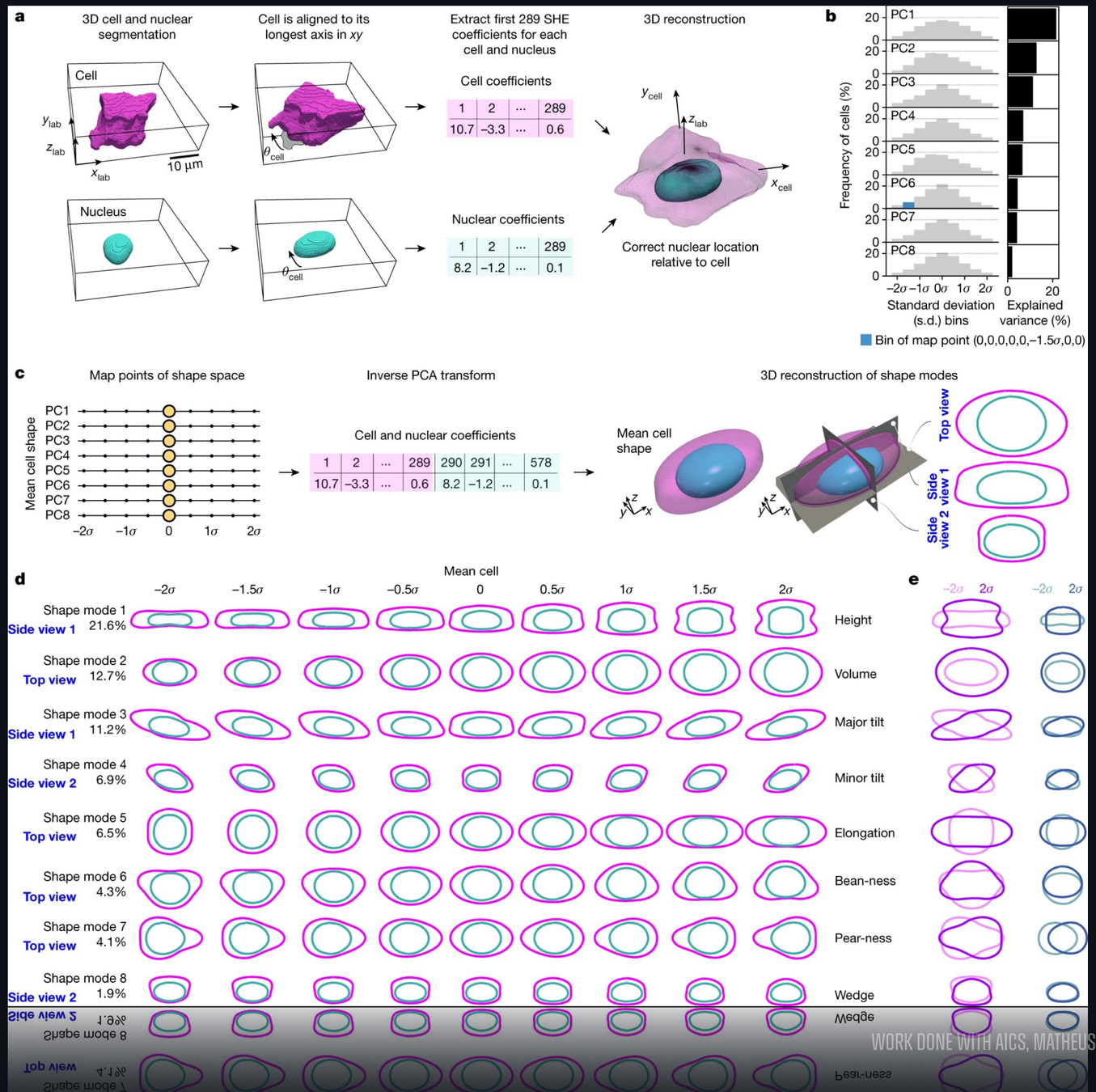


ALLEN INSTITUTE CELL SCIENCE

CVAPIPE

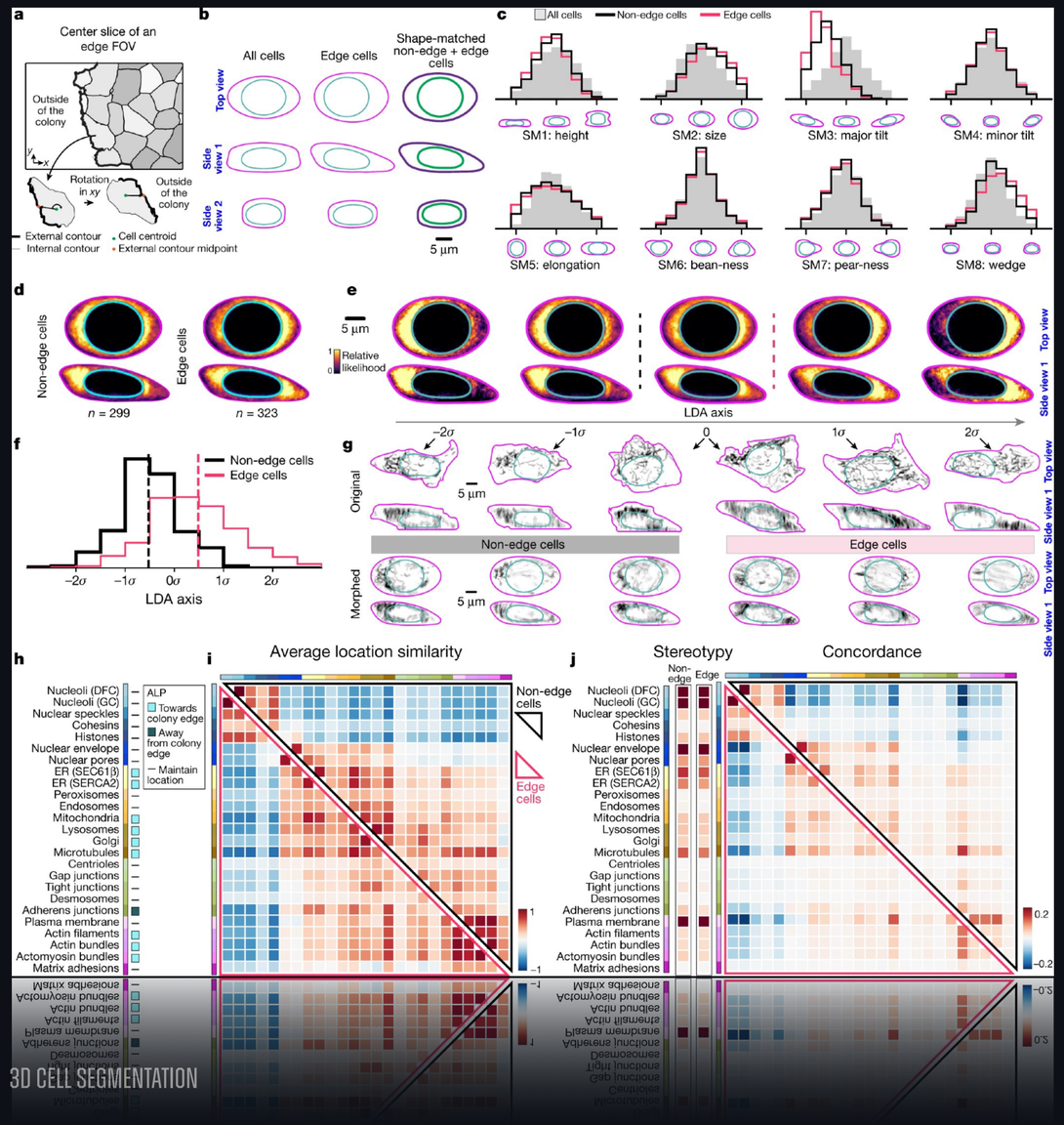
Calculating Cell Shape Modes

- Building a Cloud Pipeline
  - Build a web application that can store, analyze, and explore the CVAPIPE analysis at **petabyte scale** (Powered by AWS)
- Public Release
  - Users will be able to run the **entire** method/pipeline Matheus discussed in his talk on their own data



CVAPIPE

Calculating Cell Shape Modes



- **Pipeline Steps**
  - **Computing Single cell features**, i.e. compute the spherical harmonics coefficients for cell and nuclear shape
  - **Preprocessing** such as removing outliers and mitotic cells
  - **Computing Shapemodes** for cell and nuclear
  - **Create the parameterized intracellular location representation (PILR)**
  - **Create average PILRs**
  - **Correlate single cell PIRL**
  - **Stereotypy analysis**
  - **Concordance analysis**

Come to the Hackathon!

A web application for Cell Shape Analysis

Acknowledgments

- MATHEUS VIANA
- ALEXANDRA FERRANTE
- ALLEN INSTITUTE CELL SCIENCE (AICS) TEAM

- NINA MIOLANE
- BIOSHAPE LAB MEMBERS

- B.S. MANJUNATH (ADVISOR)
- JIAXIANG (TOM) JIANG
- VRL LAB MEMBERS

NATIONAL SCIENCE FOUNDATION SSI AWARD NO. 1664172
DDB: NSF AWARD: DGE-2125644