A general algorithm for consensus 3D cell segmentation from 2D segmented stacks

Authors

Felix Y. Zhou[1,2]
Clarence Yapp[3,4]
Zhiguo Shang[1]
Stephan Daetwyler[1,2]
Zach Marin[1,2]
Md Torikul Islam[5],
Benjamin Nanes[1,2]
Edward Jenkins[6]
Gabriel M. Gihana[1,2]
Bo-Jui Chang[1,2]
Andrew Weems[1,2]
Michael Dustin[6]
Sean Morrison[5]
Reto Fiolka[1,2],
Kevin Dean[1,2]
Andrew Jamieson[1]
Peter K. Sorger[3,4,7]
and Gaudenz Danuser[1,2]

Affiliation

[1]Lyda Hill Department of Bioinformatics, University of Texas Southwestern Medical Center, Dallas, TX, USA.
[2]Cecil H. & Ida Green Center for System Biology, University of Texas Southwestern Medical Center, Dallas, TX, USA
[3]Laboratory of Systems Pharmacology, Department of Systems Biology, Harvard Medical School, Boston, MA, 02115, USA.
[4]Ludwig Center at Harvard, Harvard Medical School, Boston, MA, 02115, USA.
[5]Children's Research Institute and Department of Pediatrics, Howard Hughes Medical Institute, University of Texas Southwestern Medical Center, Dallas, TX, USA.
[6]Kennedy Institute of Rheumatology, University of Oxford, OX3 7FY UK
[7]Department of Systems Biology, Harvard Medical School, 200 Longwood Avenue, Boston, MA 02115, USA.

Keywords

cell segmentation, instance segmentation, tissue segmentation, generalist 3D segmentation, cellular imaging

Correspondence

Correspondence to Felix Zhou (felix.zhou@utsouthwestern.edu) or Gaudenz Danuser (gaudenz.danuser@utsouthwestern.edu)

Abstract

Cell segmentation is the fundamental task. Only by segmenting, can we define the quantitative spatial unit for collecting measurements to draw biological conclusions. Deep learning has revolutionized 2D cell segmentation, enabling generalized solutions across cell types and imaging modalities. This has been driven by the ease of scaling up image acquisition, annotation and computation. However 3D cell segmentation, which requires dense annotation of 2D slices still poses significant challenges. Labelling every cell in every 2D slice is prohibitive. Moreover it is ambiguous, necessitating cross-referencing with other orthoviews. Lastly, there is limited ability to unambiguously record and visualize 1000's of annotated cells. Here we develop a theory and toolbox, u-Segment3D for 2D-to-3D segmentation, compatible with any 2D segmentation method. Given optimal 2D segmentations, u-Segment3D generates the optimal 3D segmentation without data training, as demonstrated on 11 real life datasets, >70,000 cells, spanning single cells, cell aggregates and tissue.

Main

Instance segmentation is the problem of unambiguously assigning each pixel in a 2D or voxel in a 3D image to unique objects of interest. Near universally, it is the first step in quantitative image analysis for many scientific fields including medical imaging[1] and cell biology[2]. It is only through segmentation that the objects of interest to quantify, such as nuclei[3,4], organelles[5], cells[6], bacteria[7], plants[8], organs[1,9] or vasculature[10], are explicitly identified and delineated within an image. The segmentation subsequently defines the quantitative unit of analysis to extract desired quantitative object features such as morphology[11] (e.g. length, area, and volume) and molecular expression (e.g. mean marker expression[12], subcellular patterns[13]) to perform comparative analyses or in downstream processing such as surface unwrapping[14,15].

Segmentation is easy when cells are isolated, well-contrasted and uniformly illuminated, and amenable to binary intensity thresholding and connected component analysis[16]. However, this is rare. In practice, in in-vitro culture, in-situ tissues or in-vivo, cells of diverse morphologies may interact and aggregate together in clusters that cannot be easily or accurately separated by traditional watershed techniques[2,17]. This is further compounded by inevitable variations in the imaging acquisition and staining used to visualize cellular structures resulting in weak, partial, sparse or unspecific staining of desired foreground structures[5,18].

Thanks to advancements in GPU architecture, and increased availability of publically available labelled datasets, generalist or 'foundational' 2D cell segmentation models have emerged both for interactive segmentation using prompts such as μ-SAM[19], CellSAM[20] and dense segmentation of every cell such as Cellpose[6] and various transformer models[2]. These methods leverage 'big data' and harness diversity in the training data to demonstrate impressive ability to segment 2D cells acquired across modalities and cell types[2] out-of-the box or with fine-tuning.

Physiologically, however, cells interact within complex 3D environments. The importance of studying cell biology processes in the relevant physiological 3D environments is well-documented[11,21-24]. Moreover the emergence of 3D *in-situ* tissue imaging has further provided unprecedented insights into the complex nature of the tissue microenvironment and its role in development and disease; including novel cell-cell interaction, tissue organization, and diverse cell morphologies[12,25]. Unlocking the potential of 3D imaging necessarily requires reliable, general and scalable 3D cell segmentation solutions. Simply replicating the training strategy of 2D foundation models is likely prohibitive, requiring significant amounts of well-labelled, diverse 3D cell datasets and dedicated, specialized GPU computing.

Despite the relative ease of acquisition, abundance of industrial annotation tools in 2D[26,27] and ease of crowd-sourcing and proofreading in a single field-of-view[28], the Cellpose training dataset comprises just 540 training images (total ~70,000 cells, 5 modalities) and the most recent and largest multimodal challenge[2] only 1000 training images (total 168,491 cells, 4 modalities). Replicating a densely labelled 3D dataset with comparable level of cell diversity and numbers, given more complex microenvironments, more variable image quality, and more diverse morphologies and cell packing is formidable[5,12,25,29]. Despite ongoing efforts to develop scalable 3D annotation tools[30-33] with AI assistance[34,35] and proofreading[36], it still requires significant manual expertise and intervention[37-41]. Moreover, labelling suffers from inter- and intra- annotator variation[42-44] and be biased towards easy cases. Consequently, both classical[45-48] and deep-learning based[49-51] 3D segmentation method development focus primarily on nuclei, which have well-defined round shapes, are separated from neighbors,

and visualizable with high clarity by nuclear dyes[52]. The scarce densely annotated, proofread datasets of 3D cells[53] have primarily been of plant tissue[8,54], or few cell aggregates[17,55] or of embryos[56]. Even these have few unique images and have been assembled from crops or are of different timepoints of limited true biological replicates. Synthetic[57,58], partial[59] or generative model[60,61] synthesized datasets have been proposed to alleviate the need for fully labelled data, but have so far only been demonstrated to star-convex morphologies. It is unclear how they generalize to more complex morphologies, image background and future, novel 3D imaging modalities.

High quality, annotated datasets with solid ground truth and minimal noise[62] are not the only limitation. The time to train or fine-tune foundation models is already a major consideration in 2D, requiring significant time investment, memory and specialized GPUs[1,2,19,20,63] or careful dataset curation[64]. Training comparable 3D models will not only require more time and dedicated resources, but suffer additional challenges such as model overparameterization, necessitating more efficient, revised architecture designs[10,17,65]. Lastly, even if trained on a vast dataset, foundation models still cannot guarantee generalization nor robustness[66,67]. SAM models are nevertheless fine-tuned for medical[1] and microscopy[19] and microscopy[19] datasets. Cellpose models also require retraining for best performance[68]. Moreover there is no way to choose what will work *a priori*[62]. Consequently, at the expense of reduced segmentation coverage or accuracy, it is more efficient for academic labs to adopt human-in-the-loop, interactive segmentation tools like ilastik[69] or to use segmented nuclei as seeds for 3D watershed[29].

To address the shortcomings of directly training 3D segmentation models, we revisit the idea of leveraging 2D cell segmentations to generate consensus 3D segmentations without data retraining. Using 2D predictions to assist 3D inference is common, largely to minimize computation and training. Primarily this involves adapting pretrained 2D models to 3D for example by inflating 2D convolutional kernels followed by fine-tuning in 3D[65], or are applied to process the 3D data slice-by-slice and the outputs are combined and processed by a separately trained 3D model[70]. Few works examine no-training approaches. For segmentation, near-universally 2D segmentation tools generate a 3D segmentation by matching and stitching 2D segmentations across xy slices[6,28] whereby stitching is controlled by an overlap score. Relying on a single view these 3D segmentations are notoriously rasterized and often erroneously join multiple touching cells as tubes[28,35,52,71-73]. CellStitch[74] and 3DCellComposer[52] propose matching across orthogonal xy, xz, yz views to find a consensus 3D segmentation. However these discrete matching approaches are inherently difficult to computationally scale-up with cell numbers and cannot easily handle missing, undersegmented or oversegmented cells across slices. Interestingly, Cellpose[6] proposed to average predicted 2D flow vectors along the xy, xz and yz directions to construct a 3D gradient map. By tracing the gradient map to the simulated heat origin, the 3D cell instances are found by grouping all voxels ending up in the same sink. Whilst conceptually elegant, its execution has been restricted to Cellpose predicted gradients and demonstrates limited performance on anisotropic[74], noisy or morphologically non-ellipsoidal datasets[17] despite training. We also empirically observe puzzling fragmentation artefacts around 3D cell centroids in the predicted segmentation, incompatible with its theory and inconsistent with simply stitching the equivalent cellpose 2D cell masks[74].

To derive a formal framework for 2D-to-3D segmentation unifying stitching and Cellpose proposed gradient aggregation, we revisited the instance cell segmentation problem from first principles. We find that the general 2D-to-3D aggregation can be formulated as an optimization problem, whereby we reconstruct the 3D gradient vectors of the distance transform equivalent of each cell's 3D medial-axis skeleton. This problem is then solved to obtain the corresponding 3D segmentation using gradient descent and spatial connected component analysis. To generate consensus 3D segmentations for any 2D segmentation method and using any of one, two or all three xy, xz, yz views, we developed a toolbox, u-Segment3D. u-Segment3D implements robust methods and exposes hyperparameters to flexibly handle imperfect 2D segmentations. Moreover it includes preprocessing and postprocessing methods to assist the application of pretrained models on unseen datasets, to correct and to recover missing 3D segmentation features. We first describe our formalism of 2D-to-3D segmentation. We then validate u-Segment3D by optimal, near-perfectly reconstructing the reference 3D segmentations in 11 real-life datasets, >70,000 cells from few cell aggregates, embryos, tissue, and entire vasculature networks from its 2D slice-by-slice segmentations. We then use pretrained Cellpose 2D models to demonstrate how to use u-Segment3D for any 2D method. We further validate u-Segment3D faithfully translates 2D segmentation performance and further exploits complementary information from multiple views to deliver consistent and improved 3D segmentation. In short, the better the 2D segmentation, the better the resultant 3D segmentation. Finally using pretrained Cellpose 2D models, we demonstrate the flexibility and capacity of u-Segment3D to segment unseen 3D volume data bn sets of

anisotropic cell cultures, and unwrapped embryo surfaces[15]; high-resolution single cells and cell aggregates with intricate surface protrusions[75]; thin, sprouting vasculature in zebrafish, and tissue architectures imaged with spatial multiplexing[25] and electron microscopy[76].

The u-Segment3D package is implemented in Python 3 using open-source packages. Scripts and configuration files are available to facilitate parallel computing and deployment on CPU-based high-performance computing (HPC) clusters for large 3D volumes. u-Segment3D is freely available and can be installed locally from https://github.com/DanuserLab/u-segment3D.

Results

A formal framework for 2D-to-3D segmentation

Dense instance segmentation identifies every object instance in the image and assign a unique id to all voxels comprising an instance. This is equivalent to: (i) binary labelling every image voxel as foreground (value 1) or background (value 0), and (ii) further assigning to a foreground voxel, a unique positive integer ID, (Fig. 1a). Starting with an instance segmentation of touching 2D cells, if we erode each object by 1 pixel from its border, then every cell would become spatially separated. Consequently, unique object IDs becomes redundant. The cells are equally-well represented by a binary foreground/background image, whereby object IDs are parsed by performing connected component analysis to identify spatially contiguous regions, (Fig 1b). This binary image can be factorized and equally-well computed from its 1D slices in either x- and y- directions, (Fig 1c i). Within each 1D slice we can independently identify spatially contiguous 1D regions as unique 1D 'cells', (Fig 1c ii). However, when 1D slices are restacked into a 2D image, regardless of the 1D labelling, only unique 2D objects retain spatial adjacency across 1D slices (Fig. 1c iii). Treating all 1D cells as foreground and applying connected component analysis re-identifies 2D contiguous regions and fully reconstructs the original 2D instance segmentation (Fig. 1d iv). This immediately implies that given the perfect 1D instance segmenter that accurately delineates the cell boundary even when touching in 1D, the ideal 2D segmentation can be reconstructed: run the 1D segmentation slice by slice, eroding each unique 1D 'cell' to ensure spatial separation, combine the 1D slices identified from x- and y- scans into a 2D binary image, apply spatial connected components to identify spatially contiguous 2D objects and reverse the amount eroded in 1D. Combining orthogonal views is now necessary to resolve touching cell boundaries. These first principle arguments hold equally in 3D and higher dimensions. In 3D however, instead of 1D 'slices' we have x-y, x-z, y-z 2D 'slices'. Generally, *n-1* D segmentation implies *n* D segmentation.

However, how much do we need to erode to guarantee applicability to heterogeneous cell size and morphology? Firstly, the ideal erosion process should be uniform with respect to individual cell boundary, as this ensures maximal separation from neighbors at all points on the boundary. Secondly, we observe that iterative application of uniform erosion evolves any cell shape to its medial axis skeleton[77] (MAT). At this point, every object must be spatially separated. Crucially for 2D-to-3D segmentation, the skeleton of 2D slices coincides with the medial axis skeleton of the corresponding 3D object. Resolution permitting, the 2D skeletal slices of each unique 3D object remain spatially proximal after 3D stacking, enabling identification by spatial proximity. Then, reversing the erosion will yield the 3D instance segmentation. To implement a reversible erosion process we note that the medial-axis skeletons are attractors of distance transforms[77,78], Φ and are the 'ridges' in the distance transform[78,79]. Thus uniform erosion is performed by iteratively advecting the foreground coordinates with stepsize η, in the direction of the local gradient, $\nabla\Phi$. Finally, as the distance transform is continuous and real-valued, the 3D distance transform gradients can be approximately reconstructed from its 2D slice counterparts. The general 2D-to-3D segmentation algorithm (Fig. 1d) is thus:

1. Generate all 2D segmentations in orthogonal x-y, x-z, y-z views.

2. Choose a distance transform specifying the 2D medial-axis skeleton and apply this to compute the 2D gradients for the 2D segmentations in x-y, x-z, y-z views.

3. Reconstruct the 3D gradients of the distance transform $\nabla\Phi_{xyz}^{3D}$ from the 2D gradients, using an averaging function, F

$$\text{where} \quad \nabla\Phi^{3D}_{xyz} \approx F\left(\nabla\Phi^{2D}_{xy} + \nabla\Phi^{2D}_{xz} + \nabla\Phi^{2D}_{yz}\right)$$

Use F to also reconstruct the 3D foreground binary, B

4. Identify all (x, y, z) foreground coordinates in B.

$$\text{Foreground} := \{(x^{t=0}{}_1, y_1^{t=0}, z_1^{t=0}), \ldots, (x_n^{t=0}, y_n^{t=0}, z_n^{t=0})\} \mid B(x_n^{t=0}, y_n^{t=0}, z_n^{t=0}) = 1$$

5. Apply gradient descent in 3D to iteratively propagate all foreground coordinate points for a fixed number of total iterations, T, to uncover its 3D skeleton attractor

$$(x_n^t, y_n^t, z_n^t) \leftarrow (x_n^{t-1}, y_n^{t-1}, z_n^{t-1}) - \eta\, \nabla\Phi^{3D}_{xyz}(x_n^{t-1}, y_n^{t-1}, z_n^{t-1})$$

6. Group all coordinates at final advected positions by spatial proximity and assign to each a unique positive integer object id.

$$L^{t=T}(x_n^T, y_n^T, z_n^T) = id \in \mathbb{Z}^+$$

7. Transfer the labels back from the final to the initial coordinates to obtain the 3D instance segmentation.

$$3D\ segmentation := L^{t=0}(x_n^{t=0}, y_n^{t=0}, z_n^{t=0}) = L^{t=T}(x_n^T, y_n^T, z_n^T) = id \in \mathbb{Z}^+$$

We assume throughout that gradients are unit length normalized, $\nabla\Phi \leftarrow \frac{\nabla\Phi}{|\nabla\Phi|}$. In order to operationalize this conceptual algorithm effectively for real datasets and models that predict imperfect 2D segmentations we developed a comprehensive toolbox, u-Segment3D.

u-Segment3D is a toolbox to create consensus 3D segmentations from 2D segmentations

u-Segment3D aims to execute a robust, consistent and scalable framework of generating consensus 3D segmentations given an input volume and 2D segmentations or model outputs from orthogonal views, (Fig. 1e, Suppl. Movie 1). To achieve this, u-Segment3D implements robust algorithms for each key component of the outlined general 2D-to-3D segmentation algorithm. To retain flexibility to account for imperfect 2D segmentations, algorithms expose tunable hyperparameters.

First is implementing multiple distance transforms to allow choices between speed, accuracy, and compatibility with 2D model outputs, (Extended Data Fig. 1). There is no unique definition nor computation for an object's medial-axis skeleton[77,78]. u-Segment3D considers a single centroid lying on/near the medial-axis as valid and usable for 2D-to-3D segmentation. Our implemented distance transforms are categorized into two classes; 'explicit' (Extended Data Fig. 1b) or 'implicit' (Extended Data Fig. 1c) by how their attractor is specified. Explicit transforms define explicit attractor coordinates which are incorporated as boundary conditions in computation (Methods). This ensures gradients are 0 in the attractor and enables stable convergence via gradient descent (Suppl. Movie 2). u-Segment3D implements single 'point' and multi 'point set' source attractors. The single point is the internal medial centroid, whose placement is adjustable by percentage thresholding of the cells' Euclidean distance transform (EDT) (Extended Data Fig. 1d, Methods), extending the definition in Cellpose[6]. The point set is its 2D skeleton (Methods). To compute the distance transforms, u-Segment3D considers two different partial differential equations (PDEs); the Eikonal equation which gives the geodesic solution and the Poisson equation which gives the heat diffusion solution, as used in Cellpose[6]. The Eikonal equation is faster to solve using the fast marching method[80] but generate less smooth gradients. The Poisson equation is solved exactly using LU decomposition (Methods).

Implicit transforms specify the medial skeleton implicitly as ridges. Consequently, convergence to the attractor is unstable[7,78] (Suppl. Movie 2) but is more efficient, requiring solve only the PDEs without additional constraints. The Eikonal equation can then be solved using EDT which is also an intermediary output of many 2D segmentation models[3,7]. Irrespective of the chosen distance transform, it is imperative for 2D-to-3D segmentation that the distance transform is computed in the cell to faithfully capture its shape. Iterative solutions implemented by Cellpose to solve the Poisson equation are not applicable. When diffusion is restricted in elongated and torturous structures, the gradients collapse to zero (Extended Data Figure 1e). u-Segment3D's exact solution remains robust, never zero even in very long structures, as evidence by raising the distance transform by an exponent p, Φ^p to avoid floating-point introduced non-unique gradients.

Second is implementing a content-based averaging function, F to fuse 2D image stacks, (Extended Data Fig. 2). 2D slice-by-slice segmentation may miss or under- or over-segment a cell across slices. Inspired by multiview image fusion[81,82], u-Segment3D fuses multiple image stacks using linear inverse local variance

weighting (Extended Data Fig. 2a, Methods). Using EDT as example, segmentation errors across slices cause non-continuity such that erroneous pixels have high local variance. Using inverse weighting the value of pixels from images with high-variance are down-weighted in the final fusion (Extended Data Fig. 2b). Increasing the size of the local pixel neighborhood enables correcting larger errors. For a 1x1x1 pixel neighborhood, F is the mean average fusion of Cellpose[6] and therefore no error correction. With a 5x5x5 pixel neighborhood, binary thresholding on the fused EDT perfectly recovers the foreground nuclei without artefacts (Extended Data Fig. 2c).

Third, is robustly implementing gradient descent in 2D and 3D, (Extended Data Fig. 3). For downstream spatial proximity clustering gradient descent must propagate points of the same attractor together whilst retaining spatial compactness (Extended Data Fig. 3b). We verify our implementation using a synthetic 2D image of two objects, a circle within a doughnut (Extended Data Fig. 3a). Though simple, the object gradients is complex with features typical of more nuanced morphologies such as local sinks and separating flows of opposite orientation. Running 100 iterations, whereas Cellpose implementation has orphaned foreground points of the ring and has an erroneous line attractor for the circle, u-Segment3D propagates points stably converging towards their two point attractors (Extended Data Fig. 3c) and perfect reconstruction of the original objects.

Last is implementing robust spatial proximity clustering using image-based connected component analysis, (Extended Data Fig. 4). Too many or too few clusters directly translates to over- and under- segmentation. With heterogeneity in cell shape, points will not converge to their attractors at the same time. Running gradient descent to ensure convergence for all cells is limiting in 3D. Consequently clustering must generalize to uniform point densities and irregular-shaped attractors. Adaptive local histogram thresholding used by Cellpose[6] is sensitive to point density. Density-based clustering are sensitive to the minimum number of points or radius used to define a cluster. u-Segment3D instead exploits the fact that foreground coordinates are on an image grid (Extended Data Fig. 4a). The final advected coordinates are rasterized (if floating-point) using flooring (step i). A count of the number of coordinates in each voxel is tabulated (step ii) and smoothed with a Gaussian filter of σ to build an approximate kernel density heatmap, ρ (step iii). ρ is sparse, enabling clusters represented by regional hotspots to be identified using a global threshold, $mean(\rho) + k \cdot std(\rho)$ where k can be used for adjustment. Connected component analysis labels all spatially contiguous regions with unique ids (step iv). The final segmentation is generated by indexing into this labeled image at the final advected coordinates of foreground voxels, and transferring the labeling to initial coordinates. ρ enables probabilistic cluster identification. By increasing σ u-Segment3D can 'fuzzy' link erroneously multiple clusters, equivalent to merging segmentations in the final 3D. We validated our implementation, by reconstructing the 2D cell segmentation as we propagate foreground coordinates along the gradients of the geodesic centroid distance transform (Extended Data Fig. 4b i,ii). As expected, initially (iteration 0), the segmentation is identical to applying connected component analysis to the foreground binary. As iterations increase, and attractors are found, detected cell numbers converge on the true number (Extended Data Fig. 4b iii, top). Correspondingly, segmentation quality, measure by the intersection-over-union (IoU) and F1 score, increases to 1 (Extended Data Fig. 4b iii, bottom). These observations translate also to elongated, touching cells (Extended Data Fig. 4c). Moreover, only our clustering recovers the number of clusters present in the final coordinates propagated by either Cellpose or u-Segment3D's gradient descent for the synthetic image of a circle within a doughnut (Extended Data Fig. 3d, 4d). In contrast Cellpose's clustering artificially breaks up what should be single clusters (Extended Data Fig. 3d). This critically impacts Cellpose 3D segmentation of low signal-to-noise ratio cells (Extended Data Fig. 4e,f). Whereas, Cellpose 3D grossly oversegments and fractures individual cells, u-Segment3D's gradient descent and connected component clustering recovers complete cell segmentations when applied to reparse the same predicted foreground binary and 3D gradients (Extended Data Fig. 4e,f, Suppl. Movie 3).

To maximize the utility of pretrained 2D models, u-Segment3D further implements preprocessing and postprocessing modules. The image to segment may not reflect the quality, acquisition, noise distribution and modality of the training dataset that a model was trained on. Preprocessing can help transform input images to improve performance[29,64]. However it is dataset and model-specific. Nevertheless, the following general order of processing implemented by u-Segment3D works well in practice: intensity normalization, none or any combination of denoising, deconvolution and ridge feature enhancement, and uneven illumination correction with optional gamma correction (Methods). Postprocessing follows the order of filtering out implausible segmentations based on size and consistency with the reconstructed 3D gradients (as in

Cellpose[6]), then optionally, spatial-connectivity aware label diffusion to refine segmentations to better adhere to cell boundaries within a guide image, and guided filtering to recover missing or intricate subcellular details to the individual cell segmentations. No postprocessing requires further 3D training (Methods).

The rest of the paper explores in detail each module of u-Segment3D, highlighting salient parameters and specific modifications for application to real datasets.

3D smoothing of reconstructed 3D gradients from 2D and suppressed gradient descent are essential for 2D-to-3D segmentation

To understand how the different components of the 2D-to-3D algorithm may impact 3D segmentation, we first empirically investigated 1D-to-2D segmentation of cell morphologies from the Cellpose[6] and Omnipose[7] training datasets, which can be intuitively visualized (Extended Data Fig. 5). We first examined the approximation of 2D gradients using 1D gradients (Extended Data Fig. 5a). To compute 1D gradients, we consider each disconnected 1D region as a unique 'cell'. Then, for each cell, we computed the distance of its coordinate to the slice centroid, took central differences and unit length normalized the vectors. The 2D gradients are reconstructed by stacking the x- and y- direction gradients and smoothing with a 2D Gaussian filter, width σ. We then performed gradient descent to recover the 2D segmentation and compute the gradient descent trajectory. Across single cells representing spherical, convex, branched and vessel morphologies, Gaussian filtering was essential to recover the original 2D segmentation (Extended Data Fig. 5b). With no smoothing, 1D gradients have insufficient 2D context. The reconstructed 2D gradients can fail to specify a single fixed-point attractor. Consequently trajectories do not connect all foreground pixels to a single point, unlike 2D computed gradients. 2D Gaussian filtering however restores lost 2D correlations. Increasing σ, trajectories are regularized ultimately converge to single points. The smoothing is conformalizing the initial shape, shifting its 2D centroid towards the centroid of its convex hull. For concave structures, this attractor may lie outside the cell. To examine the implications of this for gradient descent, we considered full image segmentations. For a Cellpose exemplar (90 cells), across many cells, the reconstructed 2D gradients smoothed by $\sigma = 1$ contains more than one attractor (Extended Data Fig. 6a). Consequently, after 50 iterations of gradient descent we oversegment (143 cells). Nevertheless, the reconstruction is good ($F1 = 0.77$, IoU=0.91), with the fragmentation splitting off largely small cell fragments. many cells are correctly segmented. Expectedly, increasing σ regularizes the reconstructed gradients. For $\sigma = 5$, segmentation (93 cells, F1=0.94, IoU=0.93) is on-par from ideal 2D gradients (93 cells, F1=0.98, IoU=1.00). Beyond $\sigma > 5$, gradients interact across neighboring cells, decreasing IoU and F1 performance (Extended Data Fig. 6a iv). Consequently, the number of predicted cells drops. Thus σ should be less than the smallest separation distance between the medial axis skeletons of any two cells. For long and thin tubular structures in the Omnipose exemplar (86 cells), increasing σ shifts the attractor centroid into neighboring cells. Thus increasing σ improved F1 but also decreased IoU, with an optimal balance at $\sigma = 3$ (F1=0.49, IoU=0.60) (Extended Data Fig. 6b iv). As motivated, gradient descent is only used to separate adjacent cells, not to be run to convergence. We hypothesize improved segmentations with fewer iterations. To implement this without changing total iterations, we use a variable stepsize[7], $\eta = \frac{1}{1+\tau \cdot t}$ which decays temporally with increasing iteration number t and τ adjusts the decay rate. Applying the suppressed gradient descent[7] with $\sigma = 1$ now perfectly reconstructed the 2D segmentation (86 cells, F1=1.00, IoU=1.00) (Extended Data Fig. 6c).

To test if insights from 1D-to-2D translate to 2D-to-3D we conducted the analogous reconstruction experiment for single 3D cells (Extended Data Fig. 7a). Similar to 1D-to-2D, the 2D geodesic centroid distance transform (Methods) was computed slice-by-slice in orthogonal xy, xz, yz stacks, treating spatially contiguous 2D regions as unique 'cells'. The 3D gradients was then reconstructed by averaging (F with 1x1x1 pixel neighborhood). 3D cells were selected to represent a spectrum of distinct morphologies from pseudo-spherical, to pseudo-convex and branched, and with different types of surface protrusions (Extended Data Fig. 7b). Applying suppressed gradient descent ($\tau = 0.1$) for 200 iterations, we found similar results as 1D-to-2D, with 3D cell examples of pseudo-spherical, pseudo-convex and branched morphologies with different types of surface protrusions, (Extended Data Fig. 7b,). Gaussian smoothing aids regularization and increasing σ ensures convergence to a single cell, even for the highly branched cell with filopodia ($\sigma = 15$). The same cell was fragmented into several regions at branch junctions at lower $\sigma = 1$. As expected, increasing $\tau = 0.5$ recovers perfect construction of the branched cell at the lower $\sigma = 1$.

In summary, Gaussian filtering of the reconstructed 3D gradients from 2D is key for 3D segmentation, but not implemented by Cellpose 2D. σ should be smaller than the expected minimum distance between 3D cell skeletons. To enable 3D segmentation under this limiting condition for heterogeneous morphologies, suppressed gradient descent is also essential. We next tested how different distance transforms impact 2D-to-3D segmentation in whole datasets.

u-Segment3D reconstructs the ideal 3D segmentation from ideal orthogonal 2D slice-by-slice instance segmentations

We assembled 10 published 3D datasets with dense segmentation labels and 1 additional zebrafish macrophage dataset (Suppl. Table 1). This latter dataset was curated in-house by combining connected component analysis and u-Segment3D generated segmentations[83]. DeepVesselNet[10] is a dataset of simulated binary vasculature networks. We applied connected component analysis to identify disconnected subnetworks as unique 'cells'. The total number of cells across all datasets was 73,777. For each reference cell segmentation, we extracted 8 morphological features (Fig. 2a, Methods), chosen to assess cell size (total number of voxels), the extent of elongation (stretch factor) and the topological complexity (# of skeleton nodes). To visualize in 2D the morphological diversity and variation in cell numbers across datasets, we applied UMAP[84] to the normalized features (Fig. 2b, Methods). Two plant datasets: Arabidopsis (CAM) (24,439) and Ovules (37,027) contribute the majority of the cells (83%) and dominates the UMAP. Random sampling 6 UMAP regions, the assembled datasets captures commonly found 3D morphological archetypes encountered in tissue including thin, complex vessel-like networks (Region 1), pseudo-spherical (Regions 2-4), irregular (Region 5), and tubular or branched (Region 6). Using the per dataset median UMAP coordinate, and colored UMAP by stretch factor, # skeleton nodes and volume, we broadly group the 11 datasets by the three super-morphological archetypes they best represent: complex networks (DeepVesselNet), irregular/branched (Zebrafish macrophages/Platynereis ISH nuclei/MedMNIST3D/Lateral Root Primordia), and convex (*C. Elegans* embryo/mouse organoid/mouse skull nuclei/*Platynereis* nuclei/Arabidopsis (CAM)/Ovules).

For all images in each dataset, we reconstruct the reference 3D segmentation from their ideal 2D slice-by-slice segmentations (Fig. 2c). Scanning the reference 3D segmentation in xy, xz, yz views slice-by-slice, we treat each 2D contiguous region in a 2D slice as a unique 'cell'. For each 2D 'cell', the 2D gradients is computed and used to reconstruct the 3D gradients, to generate the reconstructed 3D segmentation using 3D gradient descent and connected component analysis. Under this experiment setup, the foreground will always be correct. This allows us to unambiguously assess 3D reconstruction using 2D gradients. Three different 2D distance transforms were tested: Poisson diffusion centroid as example of an explicit transform and used in Cellpose[6]; Euclidean distance transform as example of an implicit transform and used within models like Omnipose[7], StarDist[3]; and geodesic centroid as a second example of an implicit transform, but computed differently (Methods). For all datasets, total gradient descent iterations was fixed at 250, and reference segmentations were resized to be isotropic voxels with nearest-neighbor interpolation (Suppl. Table. 1). Drawing analogy to neural network optimization, we further incorporate momentum into suppressed gradient descent to expedite convergence (Methods). Temporal decay τ was the only parameter we adjusted for each transform and dataset (Suppl. Table 2). For all, postprocessing was only to remove cells < 15 voxels. Reconstructed 3D segmentations were evaluated using the average precision (AP) curve (Methods). The AP curve reports the average fraction of cells matched between reference and predicted segmentations as the overlap cutoff (IoU) for a valid match is increased from 0.5 to 1.0 (perfect overlap) (Methods). We use the notation $\mathrm{AP}_{0.5}$ to denote AP with IoU cutoff henceforth. For perfect reconstruction, AP=1 at all IoU. In practice, due to numerical accuracy, AP always drops to 0 above an IoU cutoff.

We first analyzed the dataset from each of the three super-morphological archetypes, with the most number of cells: for convex (Ovules, Fig. 2d), for irregular (Lateral Root Primordia (LRP), Fig. 2e) and for networks (DeepVesselNet, Fig. 2f) (Suppl. Movie 5). AP curves are plotted by color for transform: magenta (geodesic), cyan (diffusion) and navy (EDT) and by marker for provided dataset split: circle (train), diamond (test) and square (validation). Impressively, we find near-perfect reconstruction across all distance transforms, morphotypes and data splits, qualitatively and quantitatively: Ovules, $\mathrm{AP}_{0.5} \approx 1$, LRP, $\mathrm{AP}_{0.5} \geq 0.8$, and DeepVesselNet, $\mathrm{AP}_{0.5} \geq 0.8$. As expected, increased τ was required for thinner, branching cells: using EDT, τ =0.5 for Ovules, and LRP, τ =2.0 for DeepVesselNet. Findings were also reflected in the other 8 datasets

(Extended Data Fig. 8), with $\mathrm{AP}_{0.5} \geq 0.75$. Moreover IoU was high, with the curve decaying prominently at IoU ≥ 0.85 and for many, IoU ≥ 0.95. IoU > 0.8 masks are near-indistinguishable from the reference by eye[7,85]. Gradient descent 2D-to-3D aggregation requires a spatial contiguous path in 3D. Consequently generated 3D segmentations naturally enforce the ideal of a single spatial component per object. This was not checked and enforced in the reference segmentation. Since we fix the foreground, the performance gap from an ideal $\mathrm{AP}_{0.5} = 1$ largely reflect inconsistent labeling in the reference 3D segmentation. For example, balanced dataset splits should exhibit the same performance. However, in LRP, the AP curve of all three transforms on the validation (val) split were notably worse. In 6/11 datasets (Ovule/Arabidopsis(CAM)/*C.Elegans*/mouse organoid/Platynereis nuclei/vesselMNIST3D), the best distance transform achieved perfect $\mathrm{AP}_{0.5} = 1.00$. Notably these are largely convex-shaped datasets or empirically observed to have images with unambiguous cell edges and minimal background. The other datasets are noisier and more morphologically complex, and thus harder to annotate and proofread in 3D. LRP is known to contain spurious labels[7,44]. For zebrafish macrophages which had the lowest maximum reconstruction performance across all datasets for a transform ($\mathrm{AP}_{0.5} = 0.8$ with Poisson), final segmentations were not rigorously proofread. Pretrained Cellpose and u-Segment3D segmentations were replaced by connected component segmentations based on automated hard-coded rules[83]. Consequently there are small and multi-component 'cells'. For DeepVesselNet, the errors are over-estimated quantitatively. The average number of subnetworks is 3, thus our results reflect on average 1 misidentified small subnetwork. Qualitatively, there is no noticeable difference in coverage (Fig. 2f), thus errors are likely data resolution-related, for example joining two subnetworks separated by a small gap (Fig. 2f, white arrow), or size filtering removed a small subnetwork or a segment connecting two subnetworks is too small and therefore unsupported during gradient computation.

Importantly we observed bias of different transforms for different morphotypes. There was minimal differences between the two explicit transforms with point-source attractors, Poisson and geodesic. However, both outperformed EDT on convex morphologies, most evidently in the LRP val (Fig. 2e), mouse skull nuclei test (Extended Data Fig 8d), Platynereis ISH nuclei test (Extended Data Fig. 8e) and zebrafish macrophages (Extended Data Fig 8h) datasets. This is primarily due to the increased stability of explicit transforms. EDT was superior for thin and complex vasculature networks (Fig. 2f, DeepVesselNet), by minimizing the distance all points needed to propagate. Overall the quantitative difference was small (<0.5 difference in $\mathrm{AP}_{0.5}$) and not as dramatic as suggested by Omnipose[7]. This is because under gradient descent the medial axis 3D skeleton is always an intermediate structure when converging towards a centroid attractor (Suppl. Movie 4). Qualitatively, we visualized both the diffusion and EDT reconstruction on exemplars from Ovules, LRP and DeepVesselNet. Despite similar F1 and IoU, only the EDT fully reconstructed all branching cells. Diffusion fragmented the cell with the longest branch (Fig. 2e, white arrows) into two 'cells'. Importantly the fragments are standalone and not erroneously part of or included parts of neighbor cells.

In summary, u-Segment3D empirically achieves near-perfect, consistent 3D segmentations from 2D slice-by-slice segmentation from orthogonal views. In the best case, we have perfect reconstruction. In the worst case, a subset of branching cells will be decomposed into a few standalone segments to be subsequently stitched. At the expense of speed, our results show the optimal distance transform applicable for all morphotypes to be explicit transforms with the 2D object skeleton defined as the attractor. This is why we have additionally implemented these in u-Segment3D (Methods). Using the 3D reconstruction results as the best upper bound of segmentation performance we next assessed the application of u-Segment3D to pretrained 2D segmentation models on the same datasets.

u-Segment3D generates consensus 3D segmentation from orthogonal 2D slice-by-slice instance segmentations of any 2D method

2D segmentation models either (i) already predict a suitable distance transform or the 2D gradients directly as an output e.g. Cellpose[6], or (ii) provides the 2D segmentation. u-Segment3D accounts for both cases (Fig. 3a). In the former, predicted 2D gradients can be used to directly generate the 3D segmentation (the direct method). In the latter, a chosen 2D distance transform is used to compute the 2D gradients from the 2D segmentations, (the indirect method). We demonstrate the pros and cons of both methods using pretrained Cellpose models. Unlike with our ideal 2D segmentations, now the reconstructed 3D foreground binary plays an additional crucial factor in performance. If the foreground does not provide a contiguous path for gradient descent, the resulting segmentation will be fragmented, even with correct gradients. For pretained Cellpose

2D models we empirically found two model parameters to crucially determine performance: (i) the diameter, which determines what-sized objects are segmented and (ii) the cell probability threshold used to determine foreground.

Cellpose 2D models already enable 'optimal' diameter prediction based on a pretrained regression model. However, this assumes one size fits all. An image can contain objects of different scales we wish to segment e.g. cell body vs cell nuclei, cells within an embryo vs the embryo shape. Moreover, a trained model is not guaranteed to generalize on out-of-sample datasets or be consistent across sequential 2D slices. When we examined Cellpose predicted cell probability and gradients on cross-sections LRP, we found seemingly similar results over a broad diameter range (Extended Data Fig.9a-c). To set diameter objectively without training, we developed an automatic tuning method based on examining the model's self-confidence. Our method runs Cellpose over a test diameter range to compute a 'contrast score' per diameter using the predicted gradients and cell probability using local pixel variance (Extended Data Fig. 10a, Methods). The resulting function uncovers all salient object scales as local maxima. Our contrast function serves as a tuning guide. Cellpos models are trained using a mean diameter of 30 pixels and documented to perform best for diameter=15-45 pixels. Based on the peak of the contrast function images are resized accordingly prior to Cellpose input. In batch operation, u-Segment3D automatically selects the optimal diameter as that which has maximum contrast. If multiple peaks are present, we can bias the selected optimal parameter to favor other maxima by adjusting the size of the considered pixel neighborhood used to compute contrast score (Extended Data Fig. 10) or by constraining the diameter range. In 3D, the cross-sectional appearance of an object can have different aspect ratios and size, even if the image is resized to be isotropic voxels. Therefore we apply our tuning to set the optimal diameter in each of xy, xz, and yz views using a representative 2D slice (Methods). As validation, the Cellpose predicted diameter matches the predicted maxima of our method (Extended Data Fig. 9a-c). Moreover, the direct method 3D segmentation using our method ($\mathrm{AP}_{0.5}$=0.28) is comparable vs using Cellpose's method ($\mathrm{AP}_{0.5}$=0.23), if not better (Extended Data Fig. 9d).

For thresholding cell probability, Cellpose does not provide automated means. u-Segment3D uses multi-threshold Otsu to statistically determine a finite number of thresholds (Methods). We can then use flooring to round thresholds to the nearest decimal point, or choose a lower threshold to strike a balance between segmentation accuracy and ensuring contiguous space for gradient descent. This works excellently for both 2D Cellpose and for 3D reconstructed cell probabilities. Given the problems we found with Cellpose's gradient descent (Extended Data Fig. 3), and spatial clustering (Extended Data Fig. 4), henceforth we always use u-Segment3D's equivalent to generate segmentations. Thus Cellpose 2D refers only to predicted gradients and cell probability outputs. Cellpose 2D segmentation refers to that after applying u-Segment3D's statistical binary thresholding, suppressed gradient descent with momentum and connected component analysis to Cellpose 2D outputs.

Using our tuning and parsing of Cellpose, with image preprocessing and segmentation postprocessing by size and gradient-consistency, we compared the direct and indirect method of u-Segment3D on 9/11 datasets (see Suppl. Table. 3 for parameter details). We excluded Zebrafish macrophages whose labels derive from u-Segment3D and VesselMNIST3D which only contains binary masks. We also considered two pretrained Cellpose models, 'cyto' and 'cyto2', both generalist models but have been trained on different datasets to assess if 2D performance translates to better 3D segmentation. To minimize data leakage, we applied models to only the validation or test splits when available.

On Ovules val split (n=2,840 cells, m=2 images, Fig. 3b i), we found excellent performance with both models (cyto: magenta line, square marker, cyto2: purple line, diamond marker) using the direct method (Fig. 3b ii, $\mathrm{AP}_{50} \approx 0.80$ for both). This was expected as the cells are convex and image quality is good, with well-defined cell edges. This is also evidenced by good performance of running Cellpose 3D mode on the same preprocessed input. Cellpose 3D does not have automatic diameter tuning and allows only one diameter across all views. To attempt as fair a comparison as possible as an end-user, without modifying the source code, we considered the oversegmentation tendency of Cellpose 3D and used the maximum of u-Segment3D inferred diameters. We also ran Cellpose 3D twice, the first to obtain 3D cell probabilities to compute the equivalent Otsu thresholds and the second to obtain final segmentations (Methods). We also used the same postprocessing parameters. As expected from algorithm design, for the same model, u-Segment3D consistently outperforms Cellpose 3D (cyto: orange line, square marker, cyto2: brown line, diamond marker). Most impressively however is that u-Segment3D even boosted the 'cyto' model to be on par with 'cyto2' from $AP_{0.5} = 0.7$ to $AP_{0.5} = 0.8$. The same was true for the test split (n=10,328 cells, m=7 images, Extended Data

Fig. 11). Again u-Segment3D boosted 'cyto' to be on par with 'cyto2' from $AP_{0.5} = 0.65$ to $AP_{0.5} = 0.7$. Compared to the best 3D construction with ideal 2D segmentations (black line, circle marker, ($\mathrm{AP}_{50} = 1.0$) however, there is a noticeable gap of 0.2. Interestingly, the indirect method with either the geodesic (magenta colored) or diffusion (cyan colored) distance transforms for both models was better quantitatively than the direct method (Fig. 3b iii, $\mathrm{AP}_{50} > 0.80$). This is likely due to better cell boundary delineation from aggregating on the hard-thresholded 2D segmentations. As compromise however total number of cells predicted is decreased (reference=1686, direct=1697, indirect=1529). Lastly, we asked if the 2D slices of the direct 3D segmentation (black lines) still retain good 2D segmentation of the image, by comparing to the native 2D slice-by-slice segmentations in xy, xz, and yz views (magenta lines) aggregated by the indirect method (Fig. 3b iv). We found that not only is the 2D segmentation preserved but also consistently improved in xy view for both models ($\mathrm{AP}_{0.5}$ = 0.45 to 0.55). This demonstrate u-Segment3D exploits complementary information from orthogonal predictions.

LRP is much more challenging, containing not only a mixture of both compact and elongated/branching cells, but cell edges are also weakly-defined (Fig. 3c). Unsurprisingly, direct u-Segment3D segmentation with both models on the val split was substantially lower ($\mathrm{AP}_{50} \approx 0.30$ for both) than that from ideal 2D segmentations ($\mathrm{AP}_{50} \approx 0.90$). This time, u-Segment3D is significantly better than Cellpose 3D ($\mathrm{AP}_{50} \approx 0.05$ for both) in both the val and test split (Extended Data Fig. 11, improving Cellpose 3D cyto ($\mathrm{AP}_{50} = 0.18$ to 0.37) and cyto2 ($\mathrm{AP}_{50} = 0.19$ to 0.40)). Interestingly the AP_{50} we measured for u-Segment3D and pretrained Cellpose were on-par reported of a Cellpose 2D (plant-cp) and Omnipose 3D (plant-omni) model trained specifically on LRP by Omnipose[7]. We thus performed a like-for-like evaluation using their pretrained model weights (Methods). Unexpectedly, plant-cp with indirect u-Segment3D and any distance transform performed best for both val ($\mathrm{AP}_{0.5}$=0.50, Extended Data Fig. 11c) and test ($\mathrm{AP}_{0.5}$=0.50-0.56) splits. Amazingly, 3D trained plant-omni and plant-cp (Cellpose 3D mode) performed only on-par with pretrained cyto2 and direct u-Segment3D in both splits. Close inspection revealed whilst plant-omni looked excellent in 3D, in cross-sectional views, it can be seen its segmentation is not complete, with many missing internal pixels. We also find plant-omni oversegments despite our additional size filtering (Methods). These results highlight the robustness of u-Segment3D and verifies we can translate better 2D models into better 3D segmentations, on-par with natively 3D trained models. Again, direct and indirect u-Segment3D segmentations were on-par in AP_{50}, but indirect is better IoU-wise, with a slightly slower drop-off (Fig. 3c iii). Again, u-Segment3D demonstrates the ability to exploit complementary information from orthogonal views. Impressively, by minimally sacrificing yz IoU, it consistently increases both xy and xz performance for both models (Fig. 3c iv).

DeepVesselNet, comprised of thin, complex vasculature networks represents the largest challenge for 2D-to-3D segmentation, (Fig. 3d). During application we found both Cellpose models predict segmentations uniformly larger than the actual vessel radii in 2D slices. Hence we additionally uniformly eroded aggregated 3D segmentations to obtain the final segmentation (Suppl. Table 3). Nevertheless there was a clear difference between the two models. Using direct segmentation, 'cyto' ($\mathrm{AP}_{0.5} = 0.5$, IoU drop-off$\approx$0.75) noticeably outperforms 'cyto2' ($\mathrm{AP}_{0.5} = 0.5$, IoU drop-off$\approx$0.75) (Fig. 3d ii). Without suppressed gradient descent, Cellpose 3D grossly oversegments ($\mathrm{AP}_{0.5} = 0$). Again, direct and indirect u-Segment3D segmentations were on-par in AP_{50}. However the indirect method is far superior in IoU, with drop-off extending to 0.95 with similar AP curves across all distance transforms (Fig. 3d iii). Again, comparing 2D segmentation performance, the direct 3D aggregated cyto outperforms individual 2D segmentations in $\mathrm{AP}_{0.5}$ but exhibits faster IoU drop-off (Fig. 3d iv). The direct aggregated cyto2 was significantly worse than its 2D counterpart. This is likely due to the 3D erosion postprocessing removing too many small 2D segmentations in slices. Since, the background appeared homogeneous in this dataset we additionally tested 2D binary Otsu thresholding. This yielded the highest $\mathrm{AP}_{0.5}$ 2D segmentations in all orthogonal views (Fig. 3d iv, green lines). Applying u-Segment3D, we consequently also achieve the highest $\mathrm{AP}_{0.5}$ 3D segmentation (Fig. 3d iii, green line).

Altogether, these three datasets, representing the three super-morphotypes of convex, irregular/branched and vessel-like, demonstrate the robust implementation and applicability of u-Segment3D to real datasets. Moreover u-Segment3D can be applied to any 2D segmentation method using the direct or indirect methods with similar $\mathrm{AP}_{0.5}$ performance. Thus we applied only the direct method on remainder datasets (Extended Data Fig. 12). Except for Arabidopsis (CAM) (best $\mathrm{AP}_{0.5} = 0.4$), whose image quality was similar to LRP and densely packed, all others had $\mathrm{AP}_{0.5} \geq 0.6$. For mouse organoids, pretrained cyto2 with u-Segment3D ($\mathrm{AP}_{0.5}$=0.93) nearly matched the ideal 2D segmentation ($\mathrm{AP}_{0.5}$=1.0).

In summary, u-Segment3D reliable 3D segmentation from 2D. Crucially the results are consistent with theoretical expectations. The better the 2D segmentation, the better the resulting 3D segmentation. However the real potential of u-Segment3D lies in its flexibility to enable segmentation of diverse unseen and out-of-distribution datasets, which do not have reference segmentations or when annotation may be ambiguous, or time-consuming to acquire.

u-Segment3D reconstructs consensus 3D segmentation from 2D slice-by-slice instance segmentations from one orthogonal view for anisotropic 3D data

Due to the microscope or culture conditions, 3D cell imaging cannot always be acquired isotropically or be interpolated to be near-isotropic later in analysis, such that image quality is similar in xy, xz, and yz. In these cases applying pretrained models like Cellpose, trained on the equivalent of in-focus 'xy' slices, to xz and yz views may yield worse segmentations. If segmenting a timelapse, we may wish to save time. In all cases, u-Segment3D can be applied without all three views. For one view, this is conceptually similar to the stitching across xy slices. Looking top-to-bottom through an epidermal organoid culture[86] (Methods), cells are initially spherical, becoming increasingly irregular and elongated (Fig. 4a). Even when interpolated to isotropic voxels, cells are still flat, and stretched in appearance (Fig. 4b). Consequently we applied Cellpose 2D to segment only xy slices, using the optimal predicted diameter of each slice. u-Segment3D then aggregated the 2D segmentations into 3D. We compared the 3D segmentation from Cellpose predicted optimal diameters (Fig. 4c,d) and from our contrast score diameters (Fig. 4e,f). Qualitatively, both look similar. Without ground-truth, and ambiguity in manual labelling without a nuclear marker, we assessed the segmentation consistency between consecutive xy slices, slice i and slice $i+1$ with $\mathrm{AP}_{0.5}$. This revealed $\mathrm{AP}_{0.5}$ variation is correlated with morphology. We find a systematic drop in $\mathrm{AP}_{0.5}$ as cell morphology changed from spherical to more elongated. Overall, our contrast score determination appears more stable, with a higher mean $\mathrm{AP}_{0.5} = 0.59$. We plotted the predicted mean cell diameter per slice (green line) with the measured cell diameter of the resultant segmentation (black line) for each method (Fig. 4g,h). Whilst Cellpose better predicts the absolute diameter per slice, their correlation across xy slices was only moderate (Pearson's R = 0.47). In contrast, u-Segment3D's contrast-score method exhibits strong correlation (Pearson's R = 0.89). Tthis consistency likely translated to the improved slice-to-slice $\mathrm{AP}_{0.5}$.

A second example is a video of MDA231 human breast carcinoma cells embedded in a collagen matrix from the single cell tracking challenge[87] (Fig. 4j). These cells have small area and thin, protrusive morphologies imaged with a noisy background. The 3D image had only 30 z slices, each cell spanning <5 slices. Again, applying pretrained Cellpose with automatic contrast-score diameter determination on xy slices only, we successfully generated consistent 3D cell segmentation. Visual inspection confirm the same cell is consistently segmented across slices. Applying our strategy to every timepoint, we also observed consistent segmentation of cells across time (Suppl. Movie 6).

Our last example is the segmentation of cells on the surface of a developing drosophila embryo from the single-cell tracking challenge[87]. Due to the curved surface, cell dynamics are better visualized using cartographic surface projections[14]. Using u-Segment3D we segmented the embryo surface (Methods) and applied u-Unwrap3D[15] to extend the cartographic projection to mapping a surface proximal subvolume (Fig. 4k). We then attempted to segment the 3D cells using xy slices only. Amazingly, despite the unequal pixel-wise metric distortion due to unwrapping, u-Segment3D still produced consistent 3D cell segmentations. This enabled us to uniquely visualize the migration (black arrows) of cells toward the ventral midline from the side in relation to cells underneath the embryo surface (Fig. 4k, Suppl. Movie 7).

u-Segment3D can refine 3D segmentations and recover subcellular detail not captured by pretrained models

In 3D, cells demonstrate a rich spectrum of protrusive, subcellular surface morphologies. Biologically, these protrusions can be classified into recurring morphological motifs[75] such as blebs, lamellipodia, filopodia, and villi. These motifs are integral to cell function. Microvilli are used by T-cells to efficiently scan target cells for antigen[88]. Blebs and lamellipodia are key for 3D migration[89,90] but have also recently been found to drive proliferation and survival in cancer cells[91,92]. These protrusions are incompletely represented in existing 2D

cell training datasets. Importantly, neural networks exhibit spectral bias[93], a phenomon of learning low-frequency modes faster and more robustly than high-frequency modes. This means that they can learn strong shape priors to predict a complete shape from hollow, partial or noisy versions. However, the recovered shape lacks intricate details such as protrusions. Rectifying this bias requires revising the architecture and additional training on fine-grained higher quality masks[94].

Active contour[55] refinement is slow, struggle with concavities[95], and cannot handle multiple cells. Random walker approaches are faster[96] but requires careful design of image forces, fine-tuning and iterations for convergence.

u-Segment3D proposes a two-stage solution to be applied after filtering out implausible cells by size and gradient consistency (Fig. 5a, i-iii). The first stage (Fig. 5a, iv) is label diffusion based on semi-supervised learning[97] to smooth and improve adherence to the cell boundaries within a guide image, whilst enforcing spatial connectivity. Each cell in the input segmentation are unique 'sources'. Each voxel simultaneously diffuses their 'source' to neighbor voxels for T iterations based on an affinity graph combining the local intensity differences in the guide image, and spatial proximity (Methods). The final segmentation is generated by assigning each voxel to the source with highest contribution (Methods). We can control the extent diffusion refines the input segmentation using a 'clamping' factor such that if 'clamped', diffusion can only modify voxels assigned to background only. We observe improved boundary matching for T<50 iterations. The guide image can be the intensity normalized raw image or any image enhancing the desired features to capture in the segmentation. The second stage uses a guide image to transfer all intricate details in the local spatial neighborhood around the diffusion-refined cell in one pass using linear-time guided filtering[98] (Methods). Conceptually, this filter is analogous to an interpolation between the binary cell mask and the intensities in the corresponding spatial region of the guide image. The neighborhood size may be fixed for all cells or set as a proportion of cell diameter. For guided filtering segmentations, we find a good image is $G = \alpha \cdot I_{norm} + (1-\alpha) I_{ridge}$, a weighted sum of the normalized input image, I_{norm} and its ridge filter-enhanced counterpart, I_{ridge}, which exaggerates subcellular protrusions.

Applying this workflow, we recovered the majority of missing surface protrusions for cell tightly packed as an aggregate whilst simultaneously retaining the benefits of the shape prior from Cellpose (Fig. 5a, Suppl. Movie 8). This meant we should be able to segment individual cells imaged with high-resolution lightsheet microscopy even when membrane staining is inhomogeneous or sparse, situations which challenge thresholding-based techniques[75]. We tested this on single cells with different morphological motifs. Knowing there is only one cell, we directly threshold the 3D reconstructed cell probability (Fig. 5b-d i). The result captures well the global morphology but cell protrusions only approximately (ii). After guided filtering, all protrusions are recovered (iii), with comparable fidelity to that of binary thresholding (iv). However, the segmentation is now much better suited for surface analysis, as measured by the genus number, g of the extracted surface mesh. The postrefined mesh has consistently lower genus than that of thresholding which were as high as $g = 91$ for filopodia. We can further recover protrusive features on touching cells in a field-of-view as shown for T-cells (Fig. 5e) and zebrafish macrophages (Fig. 5f). Lastly, as a non-cell demonstration, we tested the segmentation of zebrafish vasculature undergoing angiogenesis (Fig. 5g). The combination of using pretrained Cellpose 2D as prior and guided filtering recovered the extensive, thin sprouting vessels, despite the noisy background and inhomogeneous staining (Suppl. Table 4, Suppl. Movie 9)).

u-Segment3D can multiprocess to generate consensus 3D segmentations of tissue

3D tissues readily contain 10,000's of cells even for thin tissue of millimeter-width, sub-50 micrometer thick[25]. The time for gradient descent increases with iteration number and the number of foreground pixels (related to image size). Postprocessing increases with the number of segmented cells. To allow segmentations to be computed in a reasonable time, we also implemented a multiprocessing variant of 2D-to-3D segmentation in u-Segment3D to take advantage of the wide availability of CPU-based cluster computing (Methods). Fig. 6a illustrates the key steps, (i) the pretrained 2D model runs fast GPU inference[6,99] on 2D slices from orthogonal views; (ii, iii) gradient descent is applied in parallel to local spatially-overlapped subvolumes to generate global image coordinates. This critically ensures that border cells across subvolumes retain the same global attractor, avoiding the need to post-stitch; (iv) an existing parallelized connected component analysis

developed for large connectomics datasets is applied to generate the full image 3D segmentation; and (v) postprocessing of segmentations is applied in parallel to individual segmented cells. The segmentation of a metastatic melanoma CyCIF multiplexed tissue sample using fused nuclear and membrane signals, imaged with an equivalent isotropic voxel size 280 nm resolution (Methods), and size 194 x 5440 x 4792 pixels took ≈ 2h for preprocessing and running Cellpose slice-by-slice in xy, xz, yz, ≈ 2h to generate the initial 3D segmentation from 250 gradient descent iterations and using subvolume sizes 128 x 256 x 256 with 25% spatial overlap, ≈ 1h for size filtering and gradient consistency checking, ≈ 2h for label diffusion refinement, a total of 7h to yield the final segmentation with 43,779 cells (Suppl. Movie 10), on a CPU cluster with 32 physical cores, 72 threads, 1.5TB RAM and a single A100 GPU (40GB). Notably, the gradient descent alone would be > 20x slower. Importantly, we obtained segmentations with no stitching artifacts and agreed well with the fused cell nuclei and membrane markers when we visualize zoom-ins of the mid-slices from each of the three orthogonal views (Fig. 6a). Functionally, these segmentations enabled us to improve the accuracy of 3D cell phenotyping and to show mature and precursor T cells in metastatic melanoma engage in an unexpectedly diverse array of juxtracrine and membrane-membrane interactions[25]. Further, the extraction of 3D cell morphological features such as sphericity (Fig. 6a, region α), enabled us to reveal looser "neighbourhood" associations[100] whose morphologies reveal functional states.

Axially-swept lightsheet microscopy[101] can image thick cleared tissue volumes at subcellular resolution over thick sections up to 2mm. This enabled us to visualize single cells within micrometases in lung tissue. Unexpectedly, despite the weak fluorescence of the injected cancer cells and absence of membrane markers for cells, we could still segment all salient nuclei and micrometastases (Fig. 6c, Suppl. Movie 11). Indeed, our initial micrometastases 3D segmentation combining both channels contained many extraneous, spurious segmented cells, presumably because Cellpose attempts to also infer the shape of non-cancer cells (Extended Data Fig. 13a). Thanks to the consensus segmentation of u-Segment3D, we could nevertheless use the segmentations to measure the mean fluorescence intensity to identify well-segmented micrometases (Extended Data Fig. 13b).

Our final application was to segment cellular structures in brain tissue acquired using a recently developed technique, CATS[76] to label the extracellular compartment, and STED microscopy with tissue expansion. The result reproduces the detail within electron microscopy images whilst preserving the 3D tissue (Fig. 6d). However it is difficult to visualize any 3D structure. We applied Cellpose and u-Segment3D with the aim of an exploratory tool that 'scans' the volume to generate consensus 3D segmentations of the larger extracellular space. However these spaces are heterogeneous, different in size, and morphotype. This is challenging for Cellpose which is biased towards generating segmentations of the same scale. Applying u-Segment3D as previously was observed to fragment the thick, dominant, branching dendrites (data not shown). It is only thanks to the ability to fine-tune every step of the 2D-to-3D segmentation process in u-Segment3D that we could overwrite this bias and largely preserve the multiscale tissue architecture in the final segmentation, in 3D and in 2D cross-sections (Fig. 6d-f, Suppl. Movie 12). In particular, we fine-tuned the content-based averaging, the filtering σ to smooth 3D gradients and that used for connected component analysis.

Discussion

Here we have presented a formalism and general algorithm based on distance transforms and gradient descent to generate optimal, consensus 3D segmentations from 2D segmented volume stacks. Our formalism unifies existing works in 2D-to-3D segmentation and shows near-perfect segmentations are achievable. Conceptually our work reformulates the widespread ad hoc procedure of stitching discrete label segmentations into a continuous domain problem with controllable and easy fine-tuning. Meanwhile our work shows that the initial 2D-to-3D segmentation proposal of Cellpose, based on a point-source centroid distance transform is only a specific instance of a broader class of medial-axis distance transforms.

This led us to develop a general toolbox, u-Segment3D to robustly implement 2D-to-3D segmentation in practice for any 2D segmentation method. Through extensive validation on public datasets, we showed u-Segment3D consistently translates the performance of 2D models to 3D segmentation. The better the 2D model, the better the 3D segmentation. Moreover u-Segment3D provides fine-tuning and postprocessing method for further improving 3D segmentations. We also implemented multiprocessing to enable scalable

2D-to-3D segmentation on CPU clusters. Further speed improvements could be made such as implementing a multi-scale scheme to run u-Segment3D, which we leave for future work.

With the successes of foundation models such as ChatGPT for natural language processing and Segment Anything model in segmentation, there is a prevalent notion that everything should be learnt from data, that more data is better and models should be 'turnkey', working directly out-of-the-box or if not, be 'fine-tuned'. In the quest for generality we must not neglect the value of grounded formalism and robust design. Our analyses provide multiple cautionary tales. First, parsing the outputs of neural network models is just as important as training. By identifying and rectifying the spatial proximity clustering of Cellpose, we significantly reduced over-segmentation and boosted performance on noisy datasets. Second, considering extreme morphotypes and the simpler 1D-to-2D segmentation problem showed the critical importance of smoothing reconstructed gradients and the role of suppressed gradient descent to enable 2D-to-3D segmentation to be applied to branched and vasculature networks. Third, running Cellpose with optimal diameters in different views is necessary to capture general 3D shape, irrespective of voxel anisotropy. Because Cellpose was trained using a fixed size diameter, we could exploit Cellpose and its strong cell shape prior to 'scan' and infer all salient diameters of objects in the image. This in-turn enabled us to set optimal diameters in orthogonal views for 3D segmentation training-free. Lastly, by recognizing the spectral bias of neural networks and annotation bias, we developed simple label diffusion and guided filter postprocessing to recover intricate surface morphologies of 3D cells. This enabled us to extend neural network methods to segment high-resolution single cells where classical methods are still state-of-the-art.

In sum, our experiments question the proposition value of directly training 3D segmentation models. Using only pretrained Cellpose models equipped with automated parameter tuning, here we demonstrate an unprecedented capacity to 3D segment cells from diverse microenvironments, from single cells through to entire tissues, in-vitro, in-vivo and in-situ, and acquired from different modalities and with different resolutions. With widespread availability of diverse generalist and specialized 2D segmentation models, u-Segment3D paves a way towards accessible 3D segmentation, translating time-consuming annotation and training towards more impactful time spent on analyzing the acquired 3D datasets to provide biological insights.

Author Contributions

Conception: AJ, FYZ, GD (using pretrained 2D for 3D segmentation), FYZ (u-Segment3D); Investigation and Analysis: FYZ, CY, ZS. Data generation: CY (CyCIF tissue), SD (Zebrafish macrophages), ZM, MTI, KD, SM (lung micrometases), BN (epidermal organoid), EJ, MD (T cell co-culture), GMG, BJC (COR-L23 single cell with ruffles), AW, KD (Septin cleared tissue), RF (HBEC cell aggregate); Supervision: GD; Funding acquisition: GD, PKS, KD; Writing – Original Draft: FYZ, GD; Writing – Review and Editing: all authors.

Acknowledgments

Funding for this work in the Danuser lab was provided by the grants R35 GM136428 (NIH), and U54CA268072 (NIH). GMG is a HHMI Hanna H. Gray Fellow. EJ was funded by Wellcome Trust (grant #224040/Z/21/Z). We thank Qiongjing Zou for putting the Python library into a continuous integration coding framework. Imaging was performed using the Oxford-Zeiss Centre of Excellence in Biomedical Imaging and the Kennedy Trust for Rheumatology Research (Grant #202117 and 202103, respectively). The authors would like to acknowledge the Quantitative Light Microscopy Core, a Shared Resource of the Harold C. Simmons Cancer Center, supported in part by an NCI Cancer Center Support Grant, 1P30 CA142543-01. This research was supported in part by the computational resources provided by the BioHPC supercomputing facility located in the Lyda Hill Department of Bioinformatics, UT Southwestern Medical Center, TX. URL: https://portal.biohpc.swmed.edu. The research within this work complies with all relevant ethical regulations as reviewed and approved by the University of Texas Southwestern Medical Center. Zebrafish husbandry and experiments described here have been approved and conducted under the oversight of the Institutional Animal Care and Use Committee (IACUC) at UT Southwestern under protocol number 101805.

Data Availability

All data used in this study except for the imaging of COR-L23 ruffles, epithelial organoid, zebrafish vasculature, T cell coculture and HBEC cell aggregate are publically available from their original sources as documented in Suppl. Table 1 and in the Dataset section of Methods. All others will be made available on request to the corresponding author.

Code Availability

u-Segment3D will be available at https://github.com/DanuserLab/u-Segment3D. This Python library will also include our code to automate parameter tuning of Cellpose models.

Methods

Datasets

Validation Datasets

10 independent public datasets with reference 3D segmentations and 1 dataset collected in-house with 3D segmentation constructed with the aid of u-Segment3D were used to evaluate the ability of u-Segment3D to reconstruct 3D segmentation its ideal slice-by-slice 2D segmentations (Fig. 2) from orthogonal xy, xz, yz views. 9 of the public datasets with both images and reference segmentations were used to assess the performance of u-Segment3D with pretrained Cellpose models (Fig. 3). Details of all datasets are given in Suppl. Table 1.

Demonstration Datasets

The following datasets were collected largely in-house and 3D segmented using either the pretrained Cellpose cyto, cyto2 or nuclei 2D segmentation models, whichever qualitatively appeared to perform best with u-Segment3D (if not stated otherwise). Parameter details are provided in Suppl. Table 4.

3D epidermal organoid culture (Fig. 4a-i)

Cell culture. Human keratinocyte Ker-CT cells (ATCC #CRL-4048) were a kind gift from Dr. Jerry Shay (UT Southwestern Medical Center). A keratinocyte cell line stably expressing mNeonGreen-tagged keratin 5 (K5) to label intermediate filaments was created as previously described[86].

Epidermal organoid culture. We adapted an epidermal organoid culture model from existing protocols[86,101,102]. Polycarbonate filters with 0.4µm pore size (0.47cm^2 area, Nunc #140620) were placed in larger tissue culture dishes using sterile forceps. On day 0, a cell suspension of 5×10^5 keratinocytes expressing K5-mNeonGreen in 400µL of K-SFM was added to each filter well, and additional K-SFM was added to the culture dish to reach the level of the filter. On day 10, culture medium was aspirated from above the filter to place the cultures at an air-liquid interface. At the same time, medium in the culture dish was changed from K-SFM to differentiation medium (see Cell Culture above). On day 13, an additional 0.5 mM calcium chloride was added to the differentiation medium in the culture dish. Mature epidermal organoids were processed for imaging on day 20, after 10 days of differentiation at the air-liquid interface. Throughout the procedure, culture media were refreshed every two to three days.

Epidermal organoid imaging. Mature epidermal organoids were transferred to a clean dish, washed three times with PBS, then fixed in 4% paraformaldehyde (Electron Microscopy Sciences #15713) for 1 hour at room temperature. Filters with the organoids were cut out of the plastic housing using an 8 mm punch biopsy tool and inverted onto glass-bottom plates. Throughout imaging, PBS was added one drop at a time as needed to keep each organoid damp without flooding the dish. Organoids were imaged using a Zeiss LSM880 inverted laser scanning confocal microscope equipped with a tunable near-infrared laser for multiphoton excitation and a non-descanned detector optimized for deep tissue imaging. Images were acquired using an Achroplan 40×/0.8NA water-immersion objective resulting in an effective planar pixel size of 0.21 µm, and z-stack volumes with 1 µm step size.

Single cell tracking challenge datasets[87,103] (Fig. 4j,k)

MDA231 human breast carcinoma cells (Fluo-C3DL-MDA231) (Fig. 4j). Cells infected with a pMSCV vector including the GFP sequence, embedded in a collagen matrix captured with an Olympus FluoView F1000 microscope with Plan 20x/7 objective lens, sampling rate of 80 min and voxel size 6.0 x 1.242 x 1.242 µm.

Drosophila Melanogaster embryo (Fluo-N3DL-DRO) (Fig. 4k). Developing embryo imaged on a SIMView light-sheet microscope[104] with a sampling rate of 30s, 16x/0.8 (water) objective lens and voxel size 2.03 x 0.406 x 0.406 µm. We used Cell01 from the test dataset containing 50 timepoints. We used the pretrained Cellpose *cyto* model and u-Segment3D to segment the surface for each timepoint (Suppl. Table 3). Using the binary segmentation, we unwrapped a proximal surface depth using u-Unwrap3D[15].

Human bronchial epithelial (HBEC) cell aggregate (Fig. 5a)

Transformed HBEC cells expressing eGFP-Kras^{V12} were cultured and imaged using meSPIM and published previously[105].

Single dendritic cell with lamellipodia (Fig. 5b)

Conditionally immortalized hematopoietic precursors to dendritic cells expressing Lifeact-GFP were cultured, imaged and published previously[75].

Single HBEC cell with filopodia (Fig. 5c)

HBEC immortalized with Cdk4 and hTERT expression and transformed with p53 knockdown, Kras^{V12} and cMyc expression cultured, imaged and published previously[75].

Single COR-L23 cell with ruffles (Fig. 5d)

Culture. COR-L23 cells (Human Caucasian lung large cell carcinoma) were resuspended in 2mg/mL bovine collagen (Advanced BioMatrix 5005) and incubated for 48 hours in RPMI 1640 medium (Gibco 11875093) supplemented with 10% fetal bovine serum (PEAK SERUM PS-FB2) and 1% antibiotic-antimycotic (Gibco 14240062). Cells were incubated in a humidified incubator at 37ºC and 5% carbon dioxide.

Imaging. Images were acquired with our home-built microscope system that generates equivalents to dithered lattice light-sheet through field synthesis[106]. Briefly, the system employs a 25X NA 1.1 water immersion objective (Nikon, CFI75 Apo, MRD77220) for detection, and a 28.6X NA 0.7 water immersion (Special Optics 54-10-7) for illumination. With a 500 mm tube lens, the voxel size of the raw data is 0.104 µm × 0.104 µm × 0.300 µm. The volumetric imaging was performed by scanning the sample along the detection axis. We used a Gaussian light-sheet and optimized the light-sheet properties so the confocal length was enough to cover the cell size without sacrificing too much the axial resolution[107]. Typically, the light-sheets are about 20 µm long and 1 µm thick. In each acquisition, we optimized the laser power and the exposure time to achieve fast acquisition without introducing too much photo-bleaching. Usually, the time interval for our volumetric acquisition is chosen to be either 5 or 10 s.

T-cell coculture (Fig. 5e)

Blasted human CD8$^+$ T cells, were produced by activating naïve T cells isolated from PBMCs using anti-CD3/CD28 Dynabeads for 2 days, then rested for 5 days after removing the beads. Cells were frozen on day 5 of resting and thawed 48 hours before use. T cells were grown in complete RPMI-1640 (10% FCS, 1% Pen/Strep, 1% Glutamine, 1% HEPES) + 50 U/ml of IL-2. For migration-based imaging of multiple T cells, 8-well glass-bottom IBIDI chambers were coated with 1µg/ml of hICAM-1-6xHis linker and hCXCL11 (Peprotech) for 1 h at room temperature, washed, then coated with 1% BSA. 0.5x10^6 blasted human CD8$^+$ T cells were labelled with CellMask DeepRed diluted to a 1x working solution in imaging buffer (i.e., colourless RPMI with 1% added Pen/Strep, 1% glutamine, 1% HEPES) for 30 minutes at 37^{0}C. Cells and the glass slides were washed and resuspended in pre-warmed (37°C) imaging buffer. 0.1-0.2x10^6 Cells were gently added to the coated glass slides and left to settle for 30-minutes before imaging. Cells were imaged using the Lattice Lightsheet microscope 7 (LLSM7) from Zeiss using the 641nm laser at 4% power with 4ms of exposure. A large field of view was used for imaging multiple cells at once, with a complete volume taken every second. Deconvolution was performed using the Zeiss software.

Zebrafish macrophages (Fig. 5f)

Zebrafish larvae with fluorescent macrophages, labelled with Tg(mpeg1:EGFP) were imaged as published previously[83] (Suppl. Table 1).

Zebrafish vasculature (Fig. 5g)

Zebrafish (*Danio rerio*) embryos, larvae and adults were kept at 28.5°C and were handled according to established protocols[108,109]. All zebrafish experiments were performed at the larval stage and therefore the sex of the organism was not yet determined. To visualize the growing vasculature at around 34 h post fertilization (hpf), zebrafish larvae expressing the vascular marker *Tg(kdrl:Hsa.HRAS-mCherry)*[110] in a casper background[111] were used. To immobilize the zebrafish larvae for imaging, they were anesthetized with 200 mg/l Tricaine (Sigma Aldrich, E10521)[112] and mounted in 0.1% low melting agarose (Sigma Aldrich, A9414) inside fluorinated ethylene propylene (FEP) tubes (Pro Liquid GmbH, Art: 2001048_E; inner diameter 0.8 mm; outer diameter 1.2 mm), coated with 3% methyl cellulose (Sigma Aldrich, M0387)[113]. The mounted zebrafish larvae were imaged on a custom multi-scale light-sheet microscope with axially-swept light-sheet microscopy[83].

Multiplexed CyCIF tissue

A primary melanoma sample from the archives of the Department of Pathology at Brigham and Women's Hospital was selected. The protocol was adapted from Nirmal et al.[114]. Briefly, a fresh 35-micron thick FFPE tissue section was obtained from the block and de-paraffinized using a Leica Bond. The region in Fig. 6b was selected and annotated from a serial H&E section by board-certified pathologists as a vertical growth phase. The 35 μm thick section underwent 18 rounds of cyclical immunofluorescence (CyCIF)[115] over a region spanning 1.4 mm by 1.4 mm and sampled at 140 nm laterally and 280 nm axially. Image acquisition was conducted on a Zeiss LSM980 Airyscan 2 with a 40x/1.3NA oil immersion lens yielding a 53-plex 3D dataset[25]. A custom MATLAB script was used to register subsequent cycles to the first cycle, which was stitched in ZEN 3.9 (Zeiss). The quality of image registration was assessed with Hoechst across multiple cycles in Imaris (Bitplane). For segmentation, multiple channel markers were combined to create fused nuclei and cytoplasmic channels. Hoechst and lamin B1 were combined for nuclei. MHC-II, CD31, and CD3E were combined as a cytoplasm marker to cover all cells including tumor, blood vessels, and T cells.

Cleared tissue lung micrometastases

Cancer growth. Lung tissue containing a metastatic tumor was provided by the Morrison lab at UT Southwestern Medical Center, USA. Mice were injected with Y1.7-GFP-luciferase cells[116] and grown as previously described[117].

Lung tissue staining and clearing. Lung tissue was fixed in 4% PFA at 4^0C for less than 24 h and then washed three times with PBS with 0.02% sodium azide for 2 h per wash. The tissue was sliced into 2 mm thick sections. Tissue slices (~2mm) were permeabilized and blocked in blocking buffer (0.5% NP40, 10% DMSO, 0.5% Triton X-100, 5% donkey serum, 1X PBS) overnight at room temperature (RT). Primary and secondary antibody stock solutions were prepared in the desired concentration. Antibodies were centrifuged (MyFuge mini centrifuge) for 2 s before aliquoting in blocking buffer. Antibodies stock solutions were homogenized for at least 1 h in a shaker at 4°C before staining. Tissues were incubated in anti-GFP (1:100) for 72 h at room temperature in a tube revolver rotator. After incubation, samples were washed with wash buffer (0.5% NP40, 10% DMSO, 1X PBS) three times for 2 h each and then left rotating in wash buffer overnight. Tissues were immersed in the secondary antibody AF488 (1:250) solution for 72 h at RT. Then, secondary antibody was removed with wash buffer for at least two days changing the solution: first day 3x every 2 h and on the second day refreshed just one time. Finally, tissues were stained for nuclei with TO-PRO-3 647 (2 drops/mL) in PBS for 24 h at room temperature. Nuclear dye was washed out with wash buffer three times for 2 h each and then left rotating in wash buffer overnight. Samples were washed two times for 5 min each in PBS to remove the nuclear dye. Lung tissue was cleared using a modified iDISCO+ protocol. Lungs were dehydrated in a methanol gradient (25%/50%/75%/100%). Final clearing was achieved with repeated fresh Benzyl Alcohol and Benzyl Benzoate (BABB, 1:2) with 5g of activated aluminium oxide incubations. The samples were

washed with BABB 1:2 3x, then left standing in fresh BABB 1:2 for 15 min. Sample BABB 1:2 was refreshed and left overnight. The sample BABB 1:2 was refreshed again shortly before imaging.

Lung tissue imaging. Lung tissue slices were imaged on a ctASLMv2[118] microscope chamber controlled by navigate[119]. Nuclei were imaged using the TO-PRO-3 647 via illumination with a LuxX 642 nm, 140 mW at 100% laser power and a Semrock BLP01-647R-25 filter in the detection path. Cancer cells were imaged via illumination with a LuxX 488-150, 150 mW at 100% laser power and a Semrock FF01-515/30-32 bandpass filter in the detection path. Images were acquired with a Hamamatsu ORCA-Flash 4.0 v3 with 200 ms integration time in lightsheet readout mode.

coCATS labelled volume

We used a coCATS[76] imaging volume recorded with z-STED at near-isotropic resolution in neuropil of an organotypic hippocampal brain slice published in Michalska et al.[76] (c.f. Fig. 3). This volume was downloaded already denoised with Noise2Void.

UMAP to map morphological diversity of different cell datasets

Morphological features. Eight features were extracted for each cell based on their 3D reference segmentations.

1. *Volume* - the total number of voxels occupied by the segmented volume, calculated by binary indexing.
2. *Convexity* - the ratio of total volume to total volume occupied by the convex hull. Convex hull was computed with Python Scipy, scipy.spatial.ConvexHull on the 3D coordinates of the segmentatation volume.
3. *Major length* - The longest axial length of an ellipse fitted to the cell. Computed with Python Scikit-Image, skimage.measure.regionprops
4. *Minor length* - computed from Python Scikit-Image, skimage.measure.regionprops
5. *1 – minor length/major length* - Measure of the extent of elongation with value 0-1. When spherical, minor length = major length and the measure is 0. When very elongated, minor length << major length and the measure is 1.
6. *# skeleton segments* - Number of straight line segments the 3D binary skeleton is partitioned.
7. *# skeleton nodes* - Number of branch point nodes, where a node is defined as at least three line segments meeting at a point.
8. *Mean skeleton segment length* - Mean number of voxels in each straight line segment of the 3D binary skeleton.

The 3D binary skeleton was computed using Python Scikit-Image, skimage.morphology.skeletonize. The decomposition of the skeleton into nodes and segments was computed using the Python sknw library (https://github.com/Image-Py/sknw). Non-dimensionless measurements such as volume were not converted to metric units as only the number of raw voxels is relevant for segmentation.

UMAP parameters. The 8 features were power transformed to be more Gaussian-like using the Yeo-Johnson method[120] (Python Scikit-learn, sklearn.preprocessing.power_transform). Then z-score normalization was applied to create normalized features. Uniform Manifold Approximation and Projection (UMAP) (using the Python umap-learn library) was used to project the 8 features after normalization to 2 dimensions for visualization (n_neighbors=15, random_state=0, spread=1, metric='Euclidean'). The median UMAP coordinate for each dataset was computed by taking the median of the 2D UMAP coordinates of individual cells comprising the respective dataset. The heatmap coloring of the UMAP uses the normalized feature value and the coolwarm colorscheme clipping values to be in the range [-2,2].

u-Segment3D

u-Segment3D is a toolbox that aims to provide methods that require no further training to aggregate 2D slice-by-slice segmentations into consensus 3D segmentations. It is provided as a Python library (https://github.com/DanuserLab/u-unwrap3D). The methods within are broadly categorized into modules based on their purpose; module 1: image preprocessing; module 2: general 2D-to-3D aggregation using suppressed gradient descent with choice of different 3D distance transforms; and module 3: postprocessing to improve the concordance of segmentation to that of a guide image. Postprocessing helps achieve a tighter segmentation and recover missing local high-frequency surface protrusions.

Module 1: Preprocessing

Described below are the image preprocessing functions included in u-Segment3D to combat the primary problems of intensity normalization, image feature enhancement and uneven illumination that can greatly affect pretrained segmentation models, like Cellpose. Generally, the order of operation or the inclusion/exclusion of a step is dependent on the input data. We have found the basic workflow of i) rescaling to isotropic voxels and resizing for the desired segmentation scale, ii) uneven illumination correction, adaptive histogram equalization or gamma correction, iii) deconvolution, and iv) intensity normalization applied to the 3D raw image, to be a good starting point for Cellpose models. For Omnipose[7] models we only use intensity normalization. Any other preprocessing led to worse performance. When nuclei and cytoplasm channels present, we find Cellpose cell segmentation was better if both channels are jointly used as input.

Rescaling to isotropic voxels and resizing to the desired segmentation scale. Pretrained segmentation models work best when input images contain objects types and object sizes reflective of the original training dataset. If images are upscaled to be bigger, segmentation models may be biased towards segmenting physically smaller objects. Correspondingly if images are downscaled to be smaller, larger objects become enhanced and easier to segment as smaller objects become oversmoothed. Cellpose models are trained at a fixed diameter of 30 pixels and with isotropic xy images. We find empirically, the u-Segment3D tuning performs best for each orthoview if the input image volume is first rescaled to isotropic voxels and resized using linear interpolation so the desired feature to segment such as cell / vessel results in a peak around 30 pixels (c.f. Extended Data Fig. 9). The rescale and resize is implemented as one function using Python Scipy, scipy.ndimage.zoom function with a Python Dask tiled accelerated variant for large volumes.

Contrast enhancing intensity normalization. Image intensities are normalized such that 0 is set to the p_{lower} percentile and 1 is the p_{upper} percentile of the image intensity. By default, $p_{lower} = 2$ and $p_{upper} = 99.8$. This contrast enhances the image by clipping out sporadic high intensities caused by camera shot noise and zeroing small, but non-zero background intensities common to fluorescent microscopy.

Image deconvolution. For 2D fluorescent microscopy images or anisotropic 3D images, we use blind deconvolution with the unsupervised Wiener-Hunt approach[121] (slice-by-slice for 3D) where the hyperparameters are automatically estimated using a Gibbs sampler (implemented using Python Scikit-image, skimage.restoration.unsupervised_wiener). The initial point-spread function is specified as a 15x15 pixel sum normalized Gaussian ($\sigma = 1$) squared kernel. For 3D lightsheet imaging we use Wiener-Hunt deconvolution, with our previously published experimental PSF[75] used as a synthetic PSF.

Model-free uneven illumination correction. The raw image intensity of 2D or 3D images, I_{raw}^{ch} is corrected for uneven illumination ratiometrically, $I_{correct}^{ch} = \overline{I_{raw}^{ch}} \frac{I_{raw}^{ch}}{I_{bg}^{ch}}$ where $\overline{I_{raw}^{ch}}$ the mean image intensity of the input image and I_{bg}^{ch} is an estimate of the uneven background illumination. I_{bg}^{ch} is estimated by downsampling the image by a factor of ds, isotropic Gaussian smoothing of σ then resizing back to the dimensions of the input image. For 2D images, the downsampling factor does not need to be used and σ is specified as a fraction of the image dimension, typically 1/4 or 1/8 is a good starting point. For 3D images a default $\sigma = 5$ is used, and a

$ds = 8$ or 16. If segmentation is worse, we decrease ds by factor of 2. If $ds = 1$, Gaussian smoothing is applied at the original image resolution. The resultant enhanced image should have even illumination whilst minimal artefactual enhancement of border background intensities. A more sophisticated background correction is the N4 bias correction available in SimpleITK, originally developed for MRI image and has been successfully applied to 3D cleared-tissue imaging[29].

Adaptive histogram equalization (AHE). Contrast limited AHE or CLAHE (Python Scikit-image, skimage.exposure.equalize_adapthist) can also be used as an alternative to our model-free uneven correction. The image is divided into non-overlapping tiles and the pixel intensity is histogram equalized within each tile. Whilst this obtains good results, we find the method is computationally more memory-intensive and slower for large 3D volumes if the size of individual tiles is required to be small, thus increasing the overall number of tiles. But, there is less artefact for originally low-valued intensities compared to our fast ratiometric method.

Gamma correction. Transforms the input image, I_{in} pixelwise, raising the intensity to a power γ (float between 0-1) according so that the output image, $I_{\text{out}} = I_{in}^{\gamma}$ after scaling the image pixel intensity linearly to the range 0 to 1. Used to nonlinearly amplify low-intensity pixels to create a more uniform illumination for segmentation that is computationally inexpensive.

Vessel-like feature enhancement: Neurites, tubes, vessels, edges of cell surface protrusions all represent ridge-like structures that are both thin and long or exhibit high curvature and tortuous morphologies that are often weakly stained or visualized by raw image intensities. Ridge image filters uses the eigenvalues of the Hessian matrix of image intensities to enhance these ridge-like structures assuming the intensity changes perpendicular to but not along the structure. Many ridge filters have been developed. u-Segment3D uses the Meijering[122] filter (Python Scikit-image, skimage.filters.meijering) which enhances ridge image features of by pooling the filter responses over a list of multiple Gaussian σ. We observe empirically good performance for a diverse range of objects including vessels and cells, without requiring any other hyperparameter tuning unlike Frangi filtering[123].

Semi-automated diameter tuning for pretrained Cellpose models. The tuning process is illustrated in Extended Data Fig. 10a. Given a 2D image, the cellpose outputs, the non-normalized cell probability, p and predicted 2D gradients in x- ($\nabla_x \Phi$) and y- ($\nabla_x \Phi$) directions are computed. p is clipped to a range of [-88.72, 88.72] to avoid overflow for float32 and normalized to a value in the range [0,1], $p \leftarrow \frac{1}{1+e^{-p}}$. These ouputs are used to compute the pixelwise contrast score, $w \cdot \{\sigma_{\mathcal{N}}(\nabla_x \Phi) + \sigma_{\mathcal{N}}(\nabla_y \Phi)\}$. w is a pixelwise weight. We set $w = p$ but observe no difference if $p = 1$ for cellpose models. $\sigma(\cdot)$ is the local standard deviation at each pixel, computed over the local pixel neighborhood of width $P \times P$ pixels. The mean score over all pixels, $\frac{1}{N}\sum w \cdot \{\sigma_{\mathcal{N}}(\nabla_x \Phi) + \sigma_{\mathcal{N}}(\nabla_y \Phi)\}$ is computed over a range of equisampled diameters e.g. 15 to 120 at 2.5 increments. A centered moving average of window using symmetric padding at edges, of default 5 is used for smoothing. The result is a plot of score vs diameter. Prominent peaks in this plot highlight potential segmentations at different size scales. The more possibilities, the more peaks. Users may use this in turn to inform the diameter range to search. For automatic operation, the diameter that maximizes the contrast score is used as the optimal diameter for Cellpose. The neighborhood size functions acts like an attention mechanism (Extended Data Fig. 10c). The larger the size, the more the segmentation result corresponding to larger objects is favored. If there is no larger salient segmentation, the optimal diameter selection will be unchanged.

Semi-automated cell probability thresholding for pretrained Cellpose models. We observe for out-of-distribution images and noisy input images, pretrained Cellpose 2D models can perform well using an appropriate threshold for cell probability combined with u-Segment3D's gradient descent and spatial

connected component analysis to parse segmentations (Fig. 4j). The choice of threshold is particularly important. If the threshold is too high, there is no continuous path for the gradient descent. This results in over-segmentation. It is therefore better to veer on the side of caution and use a lower threshold to get a more connected foreground binary. However if this threshold is too low, it will not be in concordance with the predicted gradient field such that all voxels with predicted zero gradients will be extraneously and erroneously segmented. To automate the threshold, u-Segment3D applies multi-class Otsu thresholding to the normalized cell probability ($p \in [0,1]$) output of Cellpose, $p \leftarrow \frac{1}{1+e^{-p}}$. u-Segment3D further performs morphological closing to infill small holes. If only one object is known to be present, further morphological operations such as extracting the largest connected component and binary infilling can be conducted. The default Otsu thresholding is 2-class. If the segmentation partially captures the cells, we use 3-class Otsu and the lower of the two thresholds. Vice versa, if too much area is segmented, we use 3-class Otsu and the higher of the two thresholds. Optionally, we cast the threshold to the nearest decimal point, rounding down ($threshold \leftarrow \lfloor threshold * 10 \rfloor / 10$ where $\lfloor \cdot \rfloor$ is the floor operator).

Module 2: Gradient descent and distance transforms to assemble 2D slice-by-slice segmentation stacks into a 3D consensus segmentation

Methods in this module are used to implement the core 2D-to-3D segmentation algorithm outlined in Fig. 1d. If 2D segmentations are not provided in the form as a normalized cell probability (0-1) and 2D gradients in the manner of Cellpose[6], then a 2D distance transform must be used to generate the necessary 2D gradients for consensus 3D segmentation.

2D Distance transforms

u-Segment3D categorizes the distance transforms according to whether the limit or attractor of propagating points using gradient descent over an infinite number of steps is implicitly or explicitly defined (Extended Data Fig.1). Explicitly defined transforms are further categorized by the type of attractor: a single fixed point source or comprises a set of points.

u-Segment3D implements distance transforms, Φ that solve the Eikonal equation ($\|\nabla\Phi\|^2 = 1$, which gives the shortest geodesic solution) or Poisson's equation ($\nabla^2\Phi = -1$, which gives a smooth harmonic solution), for the cell interior using numerically stable methods. The Eikonal equation finds the shortest time of propagation for a point. Poisson's equation can also be viewed as solving the shortest time of propagation but with the additional constraint of minimizing curvature, yielding smoother solutions.

Implicit attractor distance transforms

With only the boundary condition $\Phi = 0$, the Eikonal and Poisson equation conceptually propagates a wave inwards symmetrically from the cell boundaries. The limit solution is the definition of the medial axis skeleton, the locus of the centers of all inscribed maximal spheres of the object where these spheres touch the boundary at more than one point[77,124,125].

Euclidean distance transform (EDT in text). Solves the Eikonal equation using fast image morphological operations. u-Segment3D uses the memory and speed optimized implementation in the Python *edt* package released by the Seung Lab (https://github.com/seung-lab/euclidean-distance-transform-3d).

Poisson distance transform. Solves the Poisson equation using LU decomposition (Python Scipy, scipy.sparse.linalg.spsolve) for each cell in an image. It is solved in parallel in u-Segment3D using the Python Dask library.

Explicit attractor distance transforms

The implicit attractor solves the equations everywhere in the cell interior. The explicit attractor variants modifies the equations to have different source terms (right hand side of equation) in different parts of the cell interior. For the Eikonal equation, $\Phi = 0$ at the cell boundary and outside, non-source points obey $\|\nabla\Phi\|^2 = 1$ and source points act as obstacles with vanishing speed, so that $\|\nabla\Phi\|^2 = 0$. For the Poisson equation, $\Phi = 0$ at the cell boundary and outside, non-source points obey the Laplace equation, $\nabla^2\Phi = 0$ and source points obey $\nabla^2\Phi = -1$.

(i) Point sources. A single interior point is designated as a point source. u-Segment3D finds the interior point with Euclidean distance transform value greater than the percentile threshold (default: 10th percentile) nearest the median coordinate of all points.

Eikonal equation solution (Geodesic centroid distance). At the interior point, $\|\nabla\Phi\|^2 = 0$. The modified equations are solved using the Fast Marching Method (FMM)[80], with the constraint enforced using a masked array by the Python scikit-fmm library. Central first order differences are used to compute the unit normalized 2D gradient.

Poisson equation solution (Poisson or diffusion centroid distance). Only at the interior point, $\nabla^2\Phi = -1$. The modified equations are solved using LU decomposition as before. To apply power transformation with exponent $p > 0$, the minimum is first subtracted from Φ to ensure positivity, $\Phi^p \coloneqq (\Phi - \Phi_{\min})^p$. Central first order differences are used to compute the respective unit normalized 2D gradient.

(ii) Point set sources. Any number of interior points are designated as point sources. u-Segment3D computes the 2D medial axis skeleton as the point set. The binary skeleton is computed from the binary cell image by iteratively removing border pixels over multiple image passes[126] (Python Scikit-image, skimage.morphology.skeletonize). This raw result can often produce skeletons that have extraneous branches that may be too close to a neighboring, contacted cell. To improve the skeleton quality, the binary image is Gaussian filtered with $\sigma = 3$ pixels, rebinarized by mean value thresholding and reskeletonized.

Eikonal equation solution (Geodesic centroid distance). For all points part of the skeleton, $\|\nabla\Phi\|^2 = 0$. The modified equations are then solved using the Fast Marching Method (FMM)[80] as above with central first order differences for computing the unit normalized 2D gradient. The gradients for all points part of the skeleton is set to zero to enforce the limiting behavior under gradient descent.

Poisson equation solution (Poisson or diffusion centroid distance). For all points part of the skeleton, $\nabla^2\Phi = -1$. The modified equations are solved using LU decomposition as above with central first order differences for computing the unit normalized 2D gradient. The gradients for all points part of the skeleton is set to zero to enforce the limiting behavior under gradient descent.

Content-based averaging function, *F*

u-Segment fuses 3D volume images, I^i from $i = 1, \dots, N$ multiple views using a content-based average function, F, with pixelwise weighting of the contribution from each view i given by the inverse local-variance, $\sigma_{\mathcal{N}}^i$ evaluated over an isotropic neighborhood, $\mathcal{N}$ of width P pixels

$$I_{fuse} = \frac{\sum_{i=1}^{N} \frac{1}{\sigma_{\mathcal{N}}^i + \alpha} I^i}{\sum_{i=1}^{N} \frac{1}{\sigma_{\mathcal{N}}^i + \alpha} + \varepsilon}$$

and α functions like a pseudo count. If α is small, $\sigma^{i}_{\mathcal{N}}$ dominates. If α is large, $\sigma^{i}_{\mathcal{N}}$ has little effect and all views are equally weighted. F is equivalent to the simple mean as used by Cellpose[6]. ε is a small value (10^{-20}) to prevent infinity. If the neighborhood is of width $P = 1$ pixels, F is also equivalent to the simple mean (Extended Data Fig. 2a). Compared to potentially more accurate approaches such as solving the multi-view reconstruction problem[82], entropy-based averaging[127] or using Gaussian filters[81], the above can be implemented more efficiently with uniform filters.

Fusing normalized 2D cell probabilities (0-1) from orthoviews and binary thresholding

Stacked normalized 2D cell probabilities (0-1) are fused using the content-based averaging function, F above with for neighborhood, $P = 1$ (default) pixels in concordance with the fusion of the 2D gradients below. For Cellpose models, the raw cell probability output, p are first clipped to the range [-88.72, 88.72] to prevent underflow/overflow in float32 and transformed, $p \leftarrow \frac{1}{1+e^{-p}}$. For methods yielding only 2D segmentations, either i) fuse using the binary then apply appropriate Gaussian filter, ii) use the intermediate cell probability image, which is always available for deep learning methods, or iii) generate a proxy cell probability image e.g. using a rescaled Euclidean distance transform.

Fusing 2D gradients from orthoviews

Stacked 2D gradients from xy, xz, yz are pre-filtered with an isotropic Gaussian of $\sigma_{pre} = 1$. The fused 3D gradients combines three separate fusion: fusing x-component from xy and xz views, y-component from xy and yz views and z-component from xz and yz views. The 3D gradients is then post-filtered with $\sigma_{post} = 1$ (default) and unit length normalized. The greater σ_{post} is, the more the regularization effect, reducing the number of attractors and preventing oversegmentation. This is helpful to segment larger and more branching structure than represented by the majority of cells using pretrained Cellpose models. However it can also merge smaller cells. For fusion, we use $\alpha = 0.5$ and in general $P = 1$ to maximize segmentation recall and perform postprocessing to remove erroneous segmentations. Larger P improves segmentation precision but may lose cells with lower contrast. These settings are generally not modified from the default. Preventing oversegmentation can be more controllably carried out by adjusting the temporal decay parameter in the gradient descent below first.

Gradient descent

Given the reconstructed 3D gradients, $\nabla\Phi$, gradient descent is applied to the set of all foreground image coordinates, $\{x_n, y_n, z_n\}$. The iterative update equation for 3D gradient descent with momentum for iteration number, $t = 0, \ldots, T$, where $T = 250$ is the total number of iterations implemented by u-Segment3D is

$$(x_n^t, y_n^t, z_n^t) \leftarrow (x_n^{t-1}, y_n^{t-1}, z_n^{t-1}) - \eta \frac{\left(\delta \cdot \nabla\Phi(x_n^{t-1}, y_n^{t-1}, z_n^{t-1}) + \mu \cdot \nabla\Phi(x_n^{t-2}, y_n^{t-2}, z_n^{t-2})\right)}{\delta + \mu}$$

Where $\nabla\Phi$ is the gradient map, μ is the momentum parameter governing the extent the past gradient is considered, ranging from 0-1 (default $\mu = 0.95$), and $\delta > \mu$ is the weighting of the current gradient and the step-size. $\mu = 0$ recovers the standard gradient descent. Nearest interpolation is used for efficiency so that (x_n^t, y_n^t, z_n^t) is always integer valued. η defines the step-size and is defined as a function of the iteration number,

$$\eta = \frac{\delta}{1 + t \cdot \tau}$$

$\tau \in \mathbb{R}^+$ is a floating point number that controls the step-size decay[7]. The greater τ is, the less the points are propagated. When $\tau = 0$, the step-size is constant $\eta = \delta$.

Parallelized variant of gradient descent on subvolumes

The volume was divided into subvolumes of (256, 512, 512) with 25% overlap. Within each subvolume we run gradient descent with momentum for 250 iterations, momenta, $\mu = 0.98$, step size $\delta = 1$ to propagate the position of foreground pixels towards its final attractor in the 3D gradient map.

Image-based connected component analysis for identifying the unique number of cell centers for instance segmentation

The method is depicted in Extended Data Fig. 4 for a 2D image and described here for a 3D image. Step (i), the final $(t = T)$ gradient descent advected foreground coordinate positions, $\{(x_n^{t=T}, y_n^{t=T}, z_n^{t=T})\}$ is rasterized onto the image grid by flooring, i.e. $\{(\lfloor x_n^{t=T} \rfloor, \lfloor y_n^{t=T} \rfloor, \lfloor z_n^{t=T} \rfloor)\}$ and clipping values to be within the bounds of the $L \times M \times N$ image volume i.e. $0 \leq \lfloor x_n^{t=T} \rfloor \leq L - 1$, $0 \leq \lfloor y_n^{t=T} \rfloor \leq M - 1$, $0 \leq \lfloor z_n^{t=T} \rfloor \leq N - 1$. Step (ii), the number of points at each voxel position is tabulated, each point contributing +1 count. Step (iii), the counts image is Gaussian filtered with $\sigma = 1$ as a fast approximation to the Gaussian kernel density estimate to produce a density heatmap, $\rho(x, y)$ for 2D and $p(x, y, z)$ for 3D. This step functions to account for uncertainty and spatially connect up points into a cluster in a soft manner assignment. The greater the Gaussian filter σ the more nearby points will be grouped into the same hotspot. This can be helpful when segmenting branching structures. (iv) The density heatmap is sparse and therefore can be segmented using a mean threshold with an optional tunable offset specified as a constant multiplicative factor, k of the standard deviation (std) of ρ, $threshold = mean(\rho) + k \cdot std(\rho)$. For real images, we set $k = 0$ and have never needed to change this. Image connected component analysis is applied to the segmented binary to create the distinct spatial cluster segmentation at $t = T$, $L^{t=T}(x, y)$ for 2D and $L^{t=T}(x, y, z)$ for 3D. Each foreground coordinate positions, $\{(x_n^{t=T}, y_n^{t=T}, z_n^{t=T})\}$ is then assigned the cell id by lookup and the final cell segmentation is computed by generating the segmentation at their initial voxel positions, $\{(x_n^{t=0}, y_n^{t=0}, z_n^{t=0})\}$. For connected component u-Segment3D uses the optimized, parallel implementation developed by the Seung Lab (https://github.com/seung-lab/connected-components-3d).

Module 3: Postprocessing the 3D consensus segmentation

Described below are the implemented postprocessing methods that can be applied to the initial 3D segmentation generated by gradient descent and connected component analysis (module 2). The recommended sequential u-Segment3D workflow is: i) removal of implausible predicted cells involving ia) removal of predicted cells below a user-specified size limit (in voxels), ib) the removal of cells which are inconsistent with that implied by the predicted 3D gradients and ic) the removal of cells that are too statistically large $(\text{volume} > \text{mean}(\text{volumes}) + k \cdot \text{std}(\text{volumes})$ where k is a multiplicative factor, default $k = 5$); ii) labelspreading to smooth, enforce the spatial connectivity constraint of segmentation and propagate segmentation to better adhere to the desired features given by a guide image; iii) guided filter refinement to transfer missing local cellular structures to the segmentation.

The guided image does not need to the same as the raw. Generally it is a version of the raw whereby desired cellular features are enhanced.

(i) Removal of implausible predicted cells.

(ia) Removal of predicted cells that are too small. Volume of individual cells are computed as number of voxels and the respective ids are removed by setting to 0 if volume less than the user-specified threshold (default 200). Additionally each cell is checked whether they are comprise multiple spatially disconnected components. If so, only the largest component is retained as each segmented cell should be spatially contiguous.

(ib) Removal of predicted cells that are inconsistent with the predicted gradients. The reconstructed 3D gradients, $\nabla\Phi_{3D\,\text{segmentation}}$ are computed from xy, xz, yz views of the assembled consensus 3D segmentation. The mean absolute error with the predicted 3D gradients is computed per cell, $MAE_{cell} = mean\left(\left|\nabla\Phi_{3D\,\text{segmentation}} - \nabla\Phi_{3D}\right|\right)_{cell}$. If $MAE_{cell} >$ user-defined threshold (default 0.8 for $\sigma_{post} = 1$). If the post Gaussian filter σ_{post} used when fusing gradients from orthoviews is >1, the threshold may need to be relaxed i.e. threshold > 0.8.

(ic) Removal of predicted cells that are statistically too large. Ratiometric uneven illumination correction may over enhance background at the borders of the image, which may result in the segmentation of very large regions. Also in dense tissue, when staining is inhomogeneous and weak, multiple closely packed cells may be segmented as one in the initial 2D segmentation. Assuming cell volumes are approximately normally distributed, we filter out improbably large cell segmentation we use mean and standard deviation (std) of all cell volumes to set a cutoff retaining all cells with volume smaller than $\mathrm{mean}(\mathrm{volume}) + k \cdot \mathrm{std}(\mathrm{cell\ volumes})$, where $k = 5$ by default.

(ii) Labelspreading to smooth and propagate cell segmentation with spatial connectivity constraint. Labelspreading[97] is a semi-supervised learning method developed to infer the label of objects in a dataset given the labels to a partial subset of the objects by diffusing on a constructed affinity graph between objects. u-Segment3D adapts this algorithm for cell segmentation. To be computationally scalable, for each cell mask, M_i, a subvolume, V_i, is cropped with the size of its rectilinear bounding box padded isotropically by a default 25 voxels. Every label in V_i is one-hot encoded to form a label vector $L \in \mathbb{R}^{N \times p}$ where N is the total number of voxels and p the number of unique cell ids, including background. We then construct an affinity matrix, A between voxels as a weighted sum ($\alpha = 0.25$) of an affinity matrix based on the intensity difference in the guide image, I between 8-connected voxel neighbors, $A_{intensity}$ and one based on the connectivity alone, $A_{laplacian}$:

$$A = \alpha A_{intensiy} + (1-\alpha) A_{laplacian} \quad \text{and}$$

$$A_{intensity}(i,j) = \begin{cases} e^{-D_{intensity}^2 \,/\, (2\mu(D_{intensity})^2)} & i \neq j \\ 1 & i = j \end{cases}$$

$$A_{laplacian}(i,j) = \begin{cases} e^{-D_{laplacian}^2 / (2\mu(D_{laplacian})^2)} & i \neq j \\ 1 & i = j \end{cases}$$

$D_{intensity}$ is the pairwise absolute difference in intensity values between two neighboring voxels i and j. $D_{laplacian}$ is the graph Laplacian with a value of 1 if a voxel i is a neighbor of voxel j, and 0 otherwise. $\mu(D)$ denotes the mean value of the entries of the matrix D. The iterative labelspreading propagation is then

$$z \in \mathbb{R}^{N \times p}$$

$$z^{t=0} = \mathbf{0}$$

$$z^{t+1} \leftarrow (1-\gamma) A\, z^t + (\gamma) L$$

where t is the interation number, $\mathbf{0}$, the empty vector and γ is a 'clamping' factor that controls the extent the original labeling is preserved. The final z is normalized using the softmax operation, and argmax is used to obtain the final cell ids. The refined cell mask, M_i^{refine} for cell id i is all voxels where the final z is assigned to the same cell id i. Parallel multiprocessing is used to efficiently apply the refinement to all individual cells. It is recommended to set the parameters per dataset, depending on the extent of correction required. We typically start with a conservative $\alpha = 0.25, \gamma = 0.75$, and run the propagation for 25 iterations. The guide image, I is usually the normalized input image (after any preprocessing) to the 2D segmentation but can be any processed image that enhances the desired cell features. For additional speed, particularly for tissue, we also typically treat each cell mask, M_i as binary instead of multi-label.

(iii) Guided image filtering to recover missing high-frequency features and subcellular protrusions. The guided filter[98], a local linear filter that can be implemented in linear time, is used to efficiently transfer the structures of a guidance image, I to the input image to be filtered, P. Setting I to be the ridge-filtered input image to enhance high-frequency cellular protrusion and vessel features, and P to be the binary mask of cell i, the resulting filtered output Q is a 'feathered' binary, being refined to appear an alpha matte near the object boundaries. The radius of the boundary that is refined is controlled by a radius parameter, $r = 35$ voxels (by default), and the extent of structure transfer by a regularization parameter, $\epsilon = 1 \times 10^{-4}$. We find the binary mask can be rough. The stronger the features are enhanced in I the more prominent the transferred structure. Q is then re-binarized using multi-threshold Otsu. Typically, we use the two-class binary Otsu. As for labelspreading, guided filtering is applied to a cropped subvolume, V_i, with the size of its rectilinear bounding box padded isotropically by a default 25 voxels. For computational efficiency, for touching cells, we perform the guided filter segmentation independently for each cell and mask out spatial regions occupied by another cell id. More accurately, we can obtain the guided filter response for all cell ids in the subvolume to define the valid region. Parallel multiprocessing is used to perform the guided filter refinement to all individual cells. Long protrusions with a length longer than r cannot be recovered using guided filtering however, the guided

filter result may assist the application of subsequent matching algorithms or serve as a improved seed image for watershed algorithms.

Semi-automatic tuning of diameter parameter in Cellpose models

The process is illustrated in Extended Data Fig. 10a. for 3D and described below.

Determining the optimal diameter for 2D image. Given a pixel neighborhood size with isotropic width, P pixels, we conduct a parameter screen of diameter = $[d_{low}, d_{high}]$ (typically $d_{low} = 10$, $d_{high} = 120$) at equal increments of 2.5 or 5. For each diameter, a contrast score is computed taking into account the 'sharpness' of the Cellpose model predicted 2D x- and y- gradients ($\nabla_x \Phi$ and $\nabla_y \Phi$ respectively) and optionally the normalized cell probability map, p (0-1).

$$\text{Contrast score}(d) = \frac{1}{N} \sum w \cdot \{\sigma_{\mathcal{N}}(\nabla_x \Phi) + \sigma_{\mathcal{N}}(\nabla_y \Phi)\}$$

Where N is the total number of image pixels, w is a pixelwise weighting set to be p and $\sigma_{\mathcal{N}}(I)$ is the pixelwise local standard deviation of the image I evaluated over the isotropic local neighborhood of width P pixels. p is computed from the unnormalized raw cell probabilities after clipping to range [-88.72, 88.72] (to prevent overflow or underflow in float32) by applying the transformation, $p \leftarrow \frac{1}{1+e^{-p}}$. The result is a contrast score function of d. A centralized moving average of 5 (if diameter increment is 2.5) or 3 (if diameter increment is 5) is applied to smooth the contrast score function. The diameter d that maximizes the contrast score is used as the optimal diameter, d_{opt} in the Cellpose model. We generally observe no difference in d_{opt} between $w = 1$ or $w = p$ for Cellpose models.

Determining the optimal diameter for 3D volume. If cell size varies slice-by-slice, the optimal diameter determination for 2D is applied slice-by-slice (Fig. 4). This becomes computationally limiting as the number of slices increase. Instead, we find good performance, if we set the optimal diameter for a representative 2D slice in each orthoview. This representative 2D slice can be set automatically to (i) the most in-focus slice as determined by the highest mean sobel magnitude, (ii) the slice with highest mean intensity, (iii) the mid-slice, or (iv) be user-defined.

Other tested segmentation methods

Cellpose 3D mode with pretrained models. We ran pretrained cellpose models in 3D mode by setting do_3D = True. Since this mode is prone to oversegmentation and Cellpose 3D only allows to have one diameter, we used the largest diameter inferred by our contrast score function. Models are run twice. The first time is to obtain the raw, unnormalized cell probability image, which was then used to determine the binarization threshold. We then ran a second time using the determined threshold to generate a 3D segmentation. We had to additionally remove all cells with volume < 2500 voxels to get the maximum average AP.

Omnipose 3D. We ran the pretrained *plant_omni* model following the example in the documentation (https://omnipose.readthedocs.io/examples/mono_channel_3D.html). This model operates on the raw image downsampled by a factor 1/3 in all dimensions and does not reinterpolate the raw image to isotropic resolution. No other preprocessing was used. We found the raw output to predict many small objects leading to an artificially low computed AP relative to qualitative assessment. Therefore we additionally removed all objects with volume < 2500 voxels to get the maximum average AP. If volumes > 2500 voxels were removed, this affected the AP computed for lateral primordial images containing smaller cells.

Cellpose 3D mode with Omnipose trained 'plant_cp' model. We ran the 2D pretrained Cellpose *plant_cp* model using the same function call as the example in the Omnipose documentation for *plant_omni* but with omni=False and do_3D=True. As for Omnipose3D we found many small predicted object and additionally postprocessed the output segmentation by removing all objects with volume < 2500 voxels.

Evaluation of segmentation quality

Segmentation quality in single images

For single 2D and 3D images, we find the optimal matching between predicted and reference cell segmentations. Given a total number of M predicted cells, and N reference cells, we iterate and find for each predicted cell i, its *K*-nearest reference cells according to the distance between their centroids. For each of the *K*-nearest reference cells, we compute the intersection-over-union (IoU) metric (0-1) (see below). This produces a $IoU(i,j) \in \mathbb{R}^{M\times K}$ matrix. We convert this to a distance cost matrix, $dist(i,j) = 1 - IoU(i,j) \in \mathbb{R}^{M\times K}$. The optimal matching between predicted and reference cells is then found by solving the linear sum assignment using a modified Jonker-Volgenant algorithm with no initialization[128] (Python Scipy, scipy.optimize.linear_sum_assignment) and retaining only the pairings that spatially overlap ($IoU(i,j) > 0$). The segmentation quality for an image is then assessed by (i) the mean IoU, to measure the spatial overlap of matched predicted and reference cells and (ii) the F1 score (see below), the harmonic mean of precision and recall to measure how accurately the segmentation detects the correct number of reference cells.

Intersection-over-union (IoU). Also called the Jaccard index, is defined as the total number of pixels in the intersection divided by the total number of pixels in the union of two binary segmentation masks A and B, $IoU(A,B) = \frac{|A\cap B|}{|A\cup B|}$.

F1 score. Predicted cells that are validly matched to a reference cell (IoU>0) define the true positives, TP. Predicted cells that are not matched are false positives, FP, and the reference cells with no valid matches are false negatives, FN. The precision is the number of matched cells divided by the total number of predicted cells, $\text{precision} = \frac{\text{TP}}{\text{TP+FP}}$.The recall is the number of matched cells divided by the total number of reference cells, $\text{recall} = \frac{\text{TP}}{\text{TP+FN}}$. F1 score is the harmonic mean of precision and recall, $\text{F1} = 2\frac{\text{precision}\times\text{recall}}{\text{precision+recall}}$.

Average precision curve

We evaluate the quality of cell segmentation using average precision consistent with popular segmentation models such as StarDist[3], Cellpose[6] and Omnipose[7]. Each predicted cell label mask is matched to the reference cell label mask that is most similar, as defined by IoU. The predictions in an image are evaluated at various levels of IoU. At a lower IoU, fewer pixels in a predicted cell have to match a corresponding reference cell for a valid match. The valid matches define the true positives, TP, the cells with no valid matches are false positives, FP, and the reference cells with no valid matches are false negatives, FN at that IoU threshold. Using these values, the standard average precision metric (AP) for each image is:

$$\text{AP} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}}$$

The average precision (AP) curve is reported for a dataset by averaging over the average precision metric for each image in the dataset. Optimal matching of predicted and reference cells is too computationally demanding in 3D even when restricting the search to nearest neighbors. We use the same approximate matching implementation in Cellpose which is derived from the fast matching functions in StarDist. It is important to note this matching is not invariant to cell id permutation. To compute the correct AP, we first relabel all cells sequentially after performing an indirect stable sort based on their (x,y,z) centroid for both reference and predicted cell segmentation independently. In line with Cellpose, the AP curve is reported for 11 IoU thresholds equisampling the range [0.5,1.0]. Many datasets e.g. Ovules do not rigorously label every cell in the image but only the cells of the primary, single connected component object in the field of view.

However pretrained Cellpose models would predict all cells in the field-of-view. For fair evaluation, for these datasets (all except for Embedseg skull nuclei, *Platynereis* nuclei and *Platynereis* ISH nuclei), we use the reference segmentation to define the foreground connected component cluster to evaluate AP and include all predicted cells within spatial connected components that have at least 25% overlap with a reference connected component cluster. For DeepVesselNet, this is at least 1% overlap due to the thinness.

Visualization

We use the Fiji ImageJ[129] 3D viewer plugin to render 3D intensity and segmentation image volumes. For visualization of intensity in Fig. 5, we acquired a snapshot of the rendering and applied an inverse lookup table. Surface meshes in Fig. 5 were extracted using u-Unwrap3D[15] and visualized using MeshLab[130]. Rotating surface mesh movies were created using ChimeraX[131].

Figures

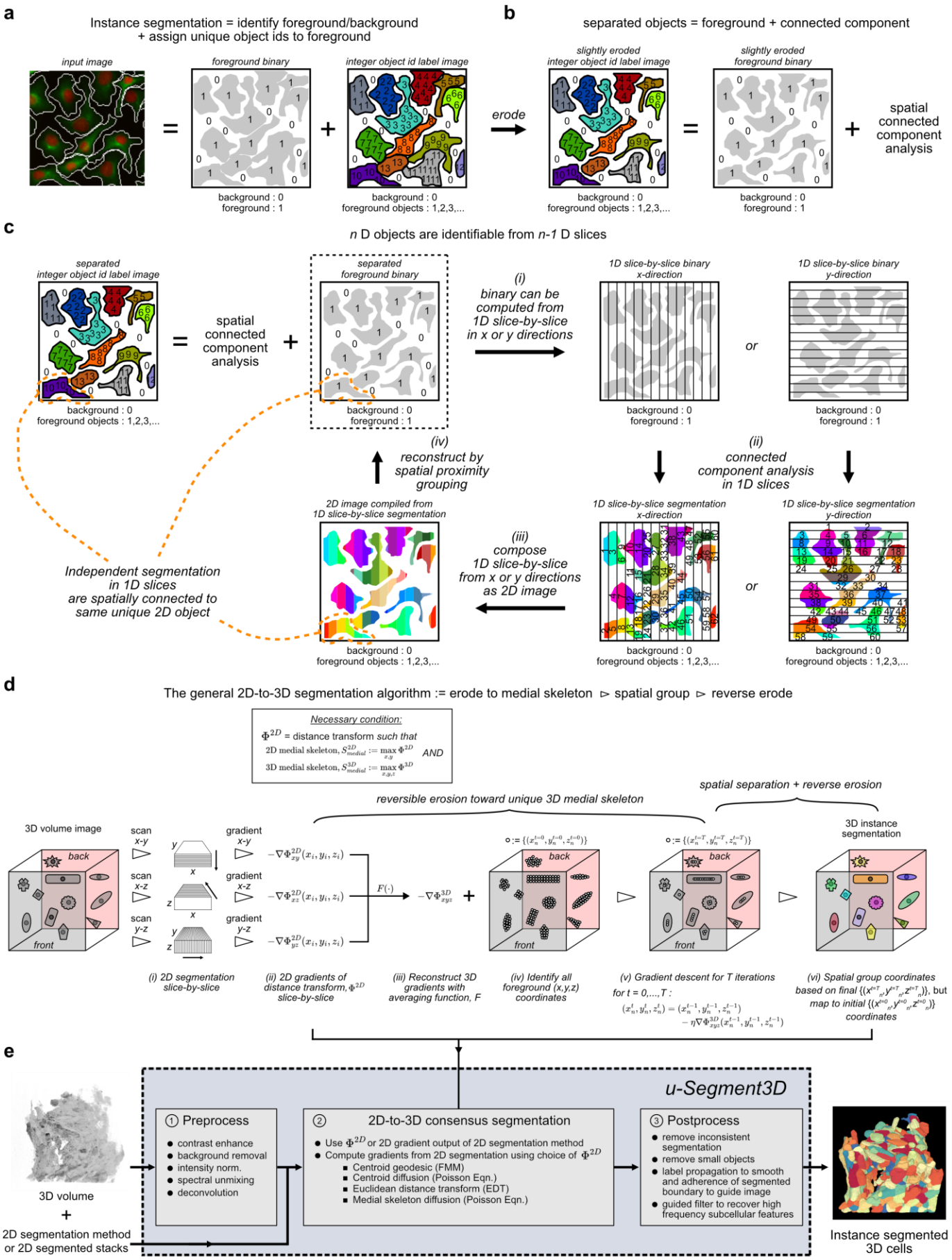


Figure 1. u-Segment3D is a toolbox for generating consensus instance 3D segmentation from 2D segmentation methods. a) Computational representation of the 2D segmentation of densely packed cells as two images, a foreground binary mask and a labelled image where each unique cell is assigned a unique integer id. **b)** Equivalent representation of the eroded segmentation such that individual cells are now spatially

separated using a single foreground binary mask which can be parsed using connected component analysis to recover individual cell ids. **c)** Schematic of the factorization of 2D instance cell segmentation to orthogonal 1D slices in *x*- or *y*- directions and subsequent perfect reconstruction from 1d instance segmentation by stacking and 2D spatial proximal grouping. **d)** Schematic of the minimal set of algorithmic steps to operationalize the conceptual framework in c) for 2D to generate the consensus 3D segmentation when cells may be densely packed. **e)** u-Segment3D is a toolbox to enable the application of the algorithmic steps in d) to real datasets with additional preprocessing methods to adapt any pretrained 2D segmentation model or 2D method and postprocessing methods to improve and recover missing local features in the reconstructed 3D segmentation such as subcellular protrusions.

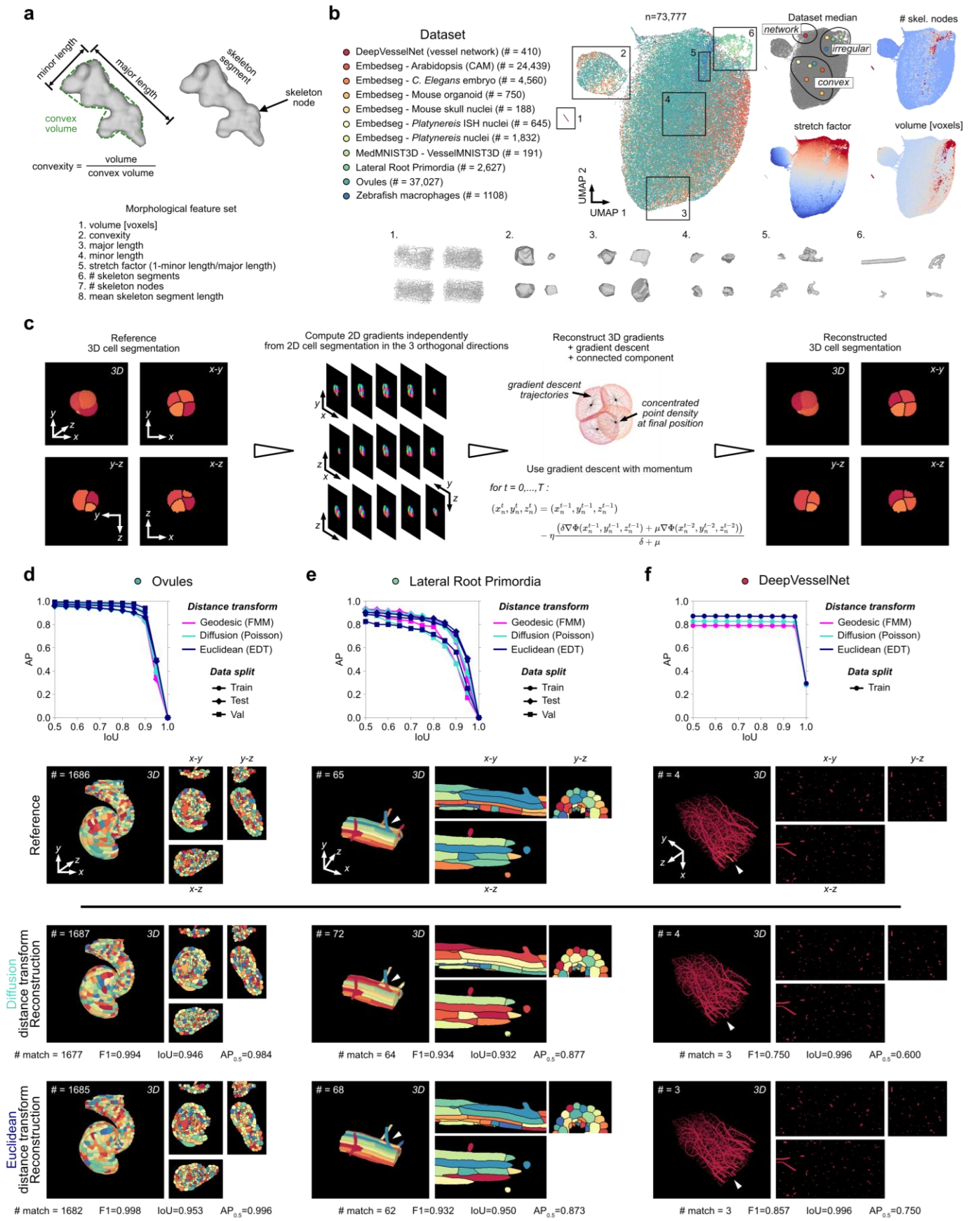


Figure 2. u-Segment3D reconstructs optimal 3D segmentation from synthetic ideal 2D segmentation labels from orthogonal x-y, x-z, y-z views. a) Illustration of the 8 computed geometrical and topological features to describe shape complexity. **b)** UMAP embedding of individual cells from 11 real datasets which together represent the spectrum of morphological complexity from convex-spherical, branching to networks. The zebrafish macrophages dataset was internally curated with the aid of u-Segment3D. All others are public (Methods). Left: colormap of individual dataset and total number of uniquely labelled cells in each dataset. Middle: UMAP, each point is a cell, color-coded by their origin dataset. Right: Median UMAP coordinate of

each dataset (top left) and heatmap of three features representing the extent of branching (total number of skeleton nodes, top right), the extent of elongation (stretch factor = 1 – minor length/major length) and their image size (total number of voxels). **c)** Illustration of the experimental workflow to compute 2D slice-by-slice distance transforms in orthogonal directions given the reference 3D cell segmentation labels and then applying u-Segment3D to reconstruct the 3D segmentation from 2D stacks for the 11 real datasets in b). **d)** Reconstruction performance measured by the average precision curve (Methods) for the Ovules dataset using three different 2D distance transforms. From top to bottom: average precision vs intersection over union (IoU) curve; 3D rendering of reference, point-based diffusion distance transform reconstructed vs skeleton-based Euclidean distance transform reconstructed 3D cell segmentation and their respective midslices in the three orthogonal views. **e), f)** Same as d) for the Lateral Root Primordia dataset containing examples of branching morphology and DeepVesselNet representing entire complex, thin network morphologies. Individual cells are uniquely colored but are not color matched with respect to the reference segmentation.

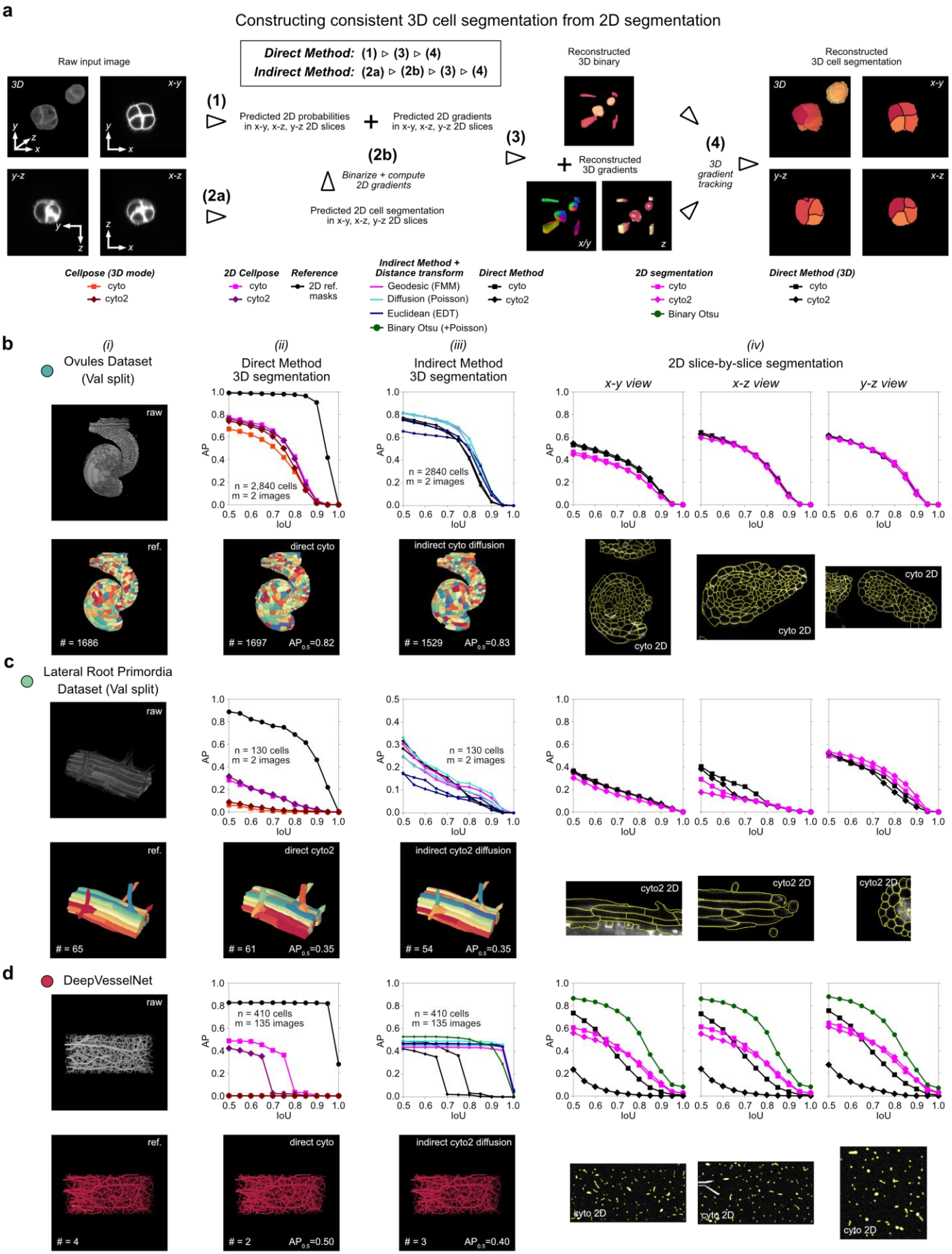


Figure 3. u-Segment3D segmentation of 3D real datasets using pretrained cellpose2D applied to orthogonal x-y, x-z, y-z views. a) Illustration of the two workflows that can be implemented by u-Segment3D to generate 3D cell segmentation. The direct method (steps 1,3,4) performs the generation utilising the 2D segmentation method's predicted distance transform or spatial gradients and cell probability maps from orthogonal views. The indirect method (steps 2a, 2b, 3, 4) first converts the stack of 2D cell segmentation

labelled images from orthogonal views using a chosen 2D distance transform to generate the necessary spatial gradients, as in Fig. 2c. **b)** 3D cell segmentation performance of the Ovules dataset (validation (val) split, n=2840 cells, m=2 volumes) using pretrained cellpose 2D with u-Segment3D. (i) 3D rendering of raw image (top) and reference 3D labels (bottom). (ii) Average precision (AP) curve for the direct method using pretrained Cellpose 2D cyto or cyto2 models relative to the AP curve of the best reconstruction from synthetic 2D segmentation in Fig. 2d (top). 3D rendering of the segmentation using the best cellpose model for the direct method (bottom). (iii) Average precision (AP) curve for the indirect method using the 2D segmentation of pretrained Cellpose 2D cyto or cyto2 models and different 2D distance transforms relative to the AP curve of the corresponding direct method constructed 3D segmentation (top). 3D rendering of the segmentation using the best cellpose model for the indirect method (bottom). (iv) Average precision (AP) curve of the 2D segmentation accuracy averaged across all 2D slices for each Cellpose 2D model (magenta lines) in each orthogonal view, x-y, x-z, y-z from left-to-right relative to the 2D segmentation accuracy of the corresponding direct method 3D segmentation (black lines). **c), d)** Same as b) for the Lateral Root Primordia (validation (val) split, n=130 cells, m=2 volumes) and DeepVesselNet (n=410 network components, m=135 images) dataset containing examples of branching morphology and DeepVesselNet representing entire complex, thin network morphologies. For d) we additionally evaluated the performance of binary Otsu thresholding as a baseline 2D segmentation method (green line).

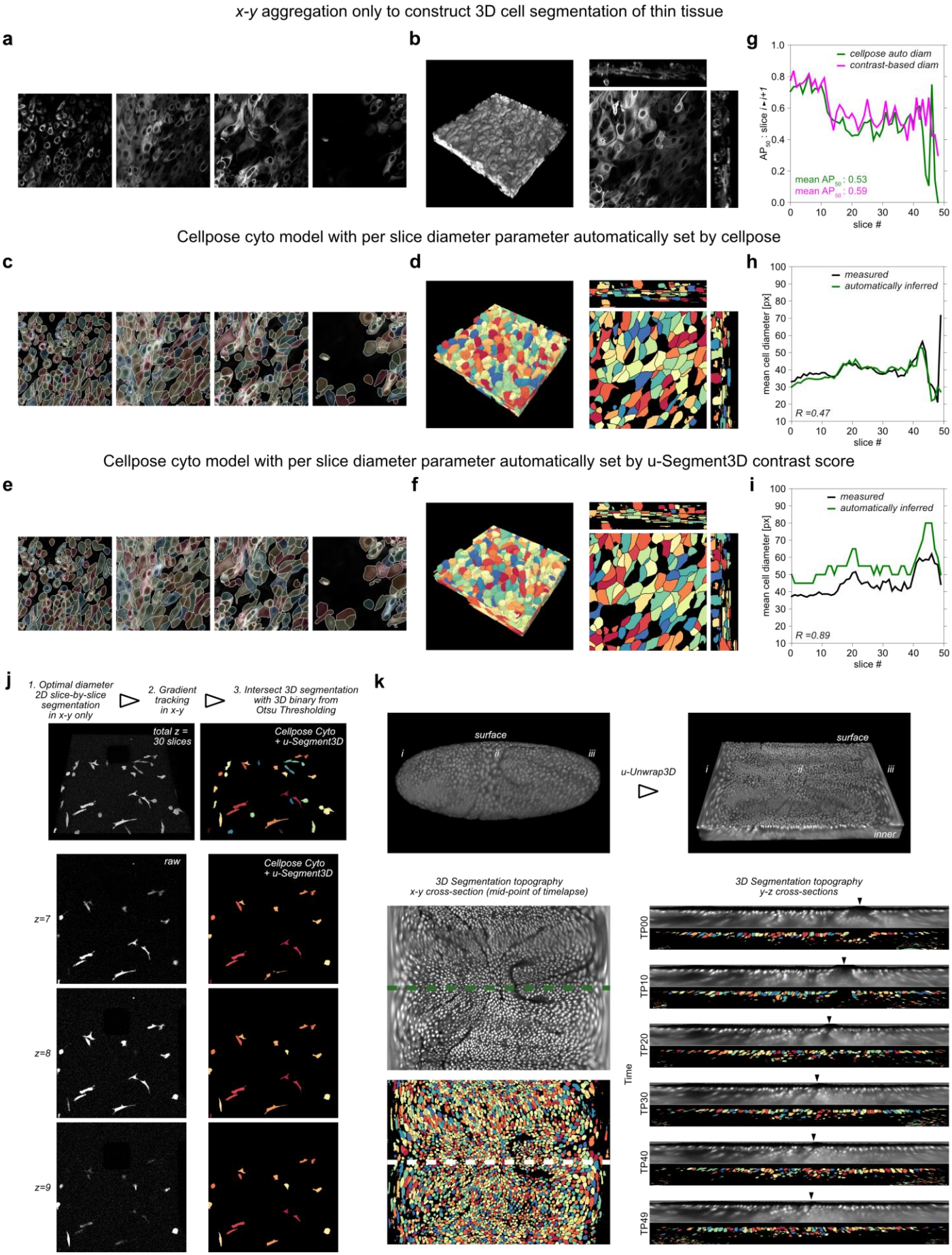


Figure 4. u-Segment3D segmentation of anisotropic 3D volumes using only x-y 2D stacks. a) Four equi-sampled x-y image slices from top-to-bottom of the air-liquid interface keratinocyte culture. **b)** 3D render of the air-liquid interface keratinocyte culture with axial interpolation to isotropic voxel resolution (left) and corresponding mid-section orthoslices (right). **c)** Cellpose 2D cell segmentations using the 'cyto' model and diameter automatically determined per-slice by cellpose. Cells are individually colored and overlaid onto the

four x-y image slices in a). White boundaries delineate individual cell boundaries within a slice. **d)** 3D render of the u-Segment3D consensus segmentation of the x-y 2D segmentation stacks in c) (left) and corresponding mid-section orthoslices (right). **e)** Cellpose 2D cell segmentations using the 'cyto' model and diameter automatically determined per-slice by u-Segment3D contrast score. Cells are individually colored and overlaid onto the four x-y image slices in a). White boundaries delineate individual cell boundaries within a slice. **f)** 3D render of the u-Segment3D consensus segmentation of the x-y 2D segmentation stacks in e) (left) and corresponding mid-section orthoslices (right). **g)** 2D cell segmentation consistency measured by the average precision at IoU cutoff = 0.5 between success z-slices as a function of z-slice id for per-slice Cellpose model diameter auto-determined by Cellpose (green line) or u-Segment3D contrast score (magenta line). **h)** Mean cell diameter inferred by Cellpose (green line) and measured after obtaining the corresponding 2D cell segmentation (black line) for each xy-slice. **i)** Mean cell diameter inferred by peak position in the u-Segment3D contrast score (green line) and measured after obtaining the corresponding 2D cell segmentation (black line) for each xy-slice. **j)** Segmentation of MDA231 human breast carcinoma cells from the 3D Cell Tracking Challenge using u-Segment3D to aggregate Cellpose 2D xy-slice segmentations with optimal diameter selection by contrast score. 3D render of raw and 3D cell segmentation (top) and in consecutive 2D xy-slices (bottom). **k).** u-Segment3D per-frame segmentation of unwrapped proximal surface topography volumes of drosophila. i) Unwrapping of the proximal surface using u-Unwrap3D. ii) Cellpose 2D and u-Segment3D contrast score diameter segmentation of the surface x-y slice at timepoint (TP) 25. iii) Mid y-z cross-section snapshots of the raw (top) and segmented (bottom) topography volumes for 6 equi-spaced timepoints. *R* denotes the Pearson's *R* in panels g)-i).

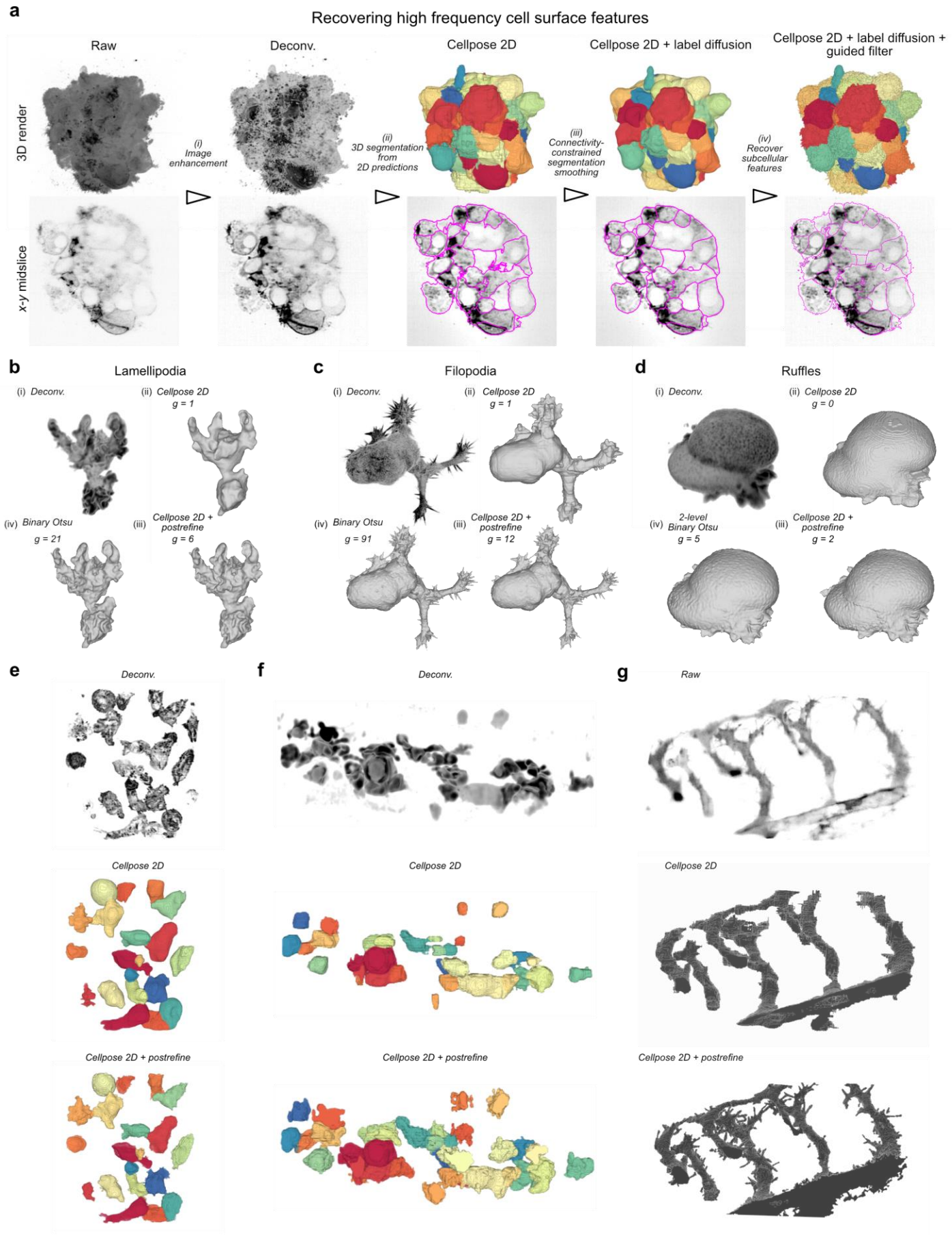


Figure 5. u-Segment3D postprocessing recovers missing high-frequency, high-curvature subcellular features. a) General u-Segment3D workflow with postprocessing to segment individual cells and recover subcellular features of each cell. 3D render (top) and x-y midslice (bottom) of the output at each step. **b)** Binary segmentation and recovery of lamellipodial features on a dendritic cell using u-Segment3d postprocessing. 3D rendering of the (i) deconvolved input, (ii) initial 3D segmentation from aggregated cellpose 2D cell probability map (after step ii of a)), (iii) final postprocessed 3D segmentation (after step iv of

a)) and comparison with the segmentation from binary Otsu thresholding on the 3D image intensity. *g* = genus of extracted surface mesh. **c), d)** Binary segmentation and recovery of filopodial and ruffle features on a HBEC and COR-L23 cell using u-Segment3d postprocessing. **e)** Single cell 3D segmentation of T-cells using cellpose 2D with u-Segment3D postprocessing. 3D render of deconvolved image volume (top), initial 3D segmentation from aggregated cellpose 2D (middle) and final 3D segmentation with recovered subcellular protrusions (bottom). **f)** Single cell 3D segmentation of zebrafish macrophages using cellpose 2D with u-Segment3D postprocessing. 3D render of deconvolved image volume (top), initial 3D segmentation from aggregated cellpose 2D (middle) and final 3D segmentation with recovered subcellular protrusions (bottom). **g)** Binary 3D segmentation of developing zebrafish vasculture using cellpose 2D with u-Segment3D postprocessing. 3D render of raw image volume (top), initial 3D segmentation from aggregated cellpose 2D cell probability maps (middle) and final 3D segmentation with recovered sprouting vessels (bottom).

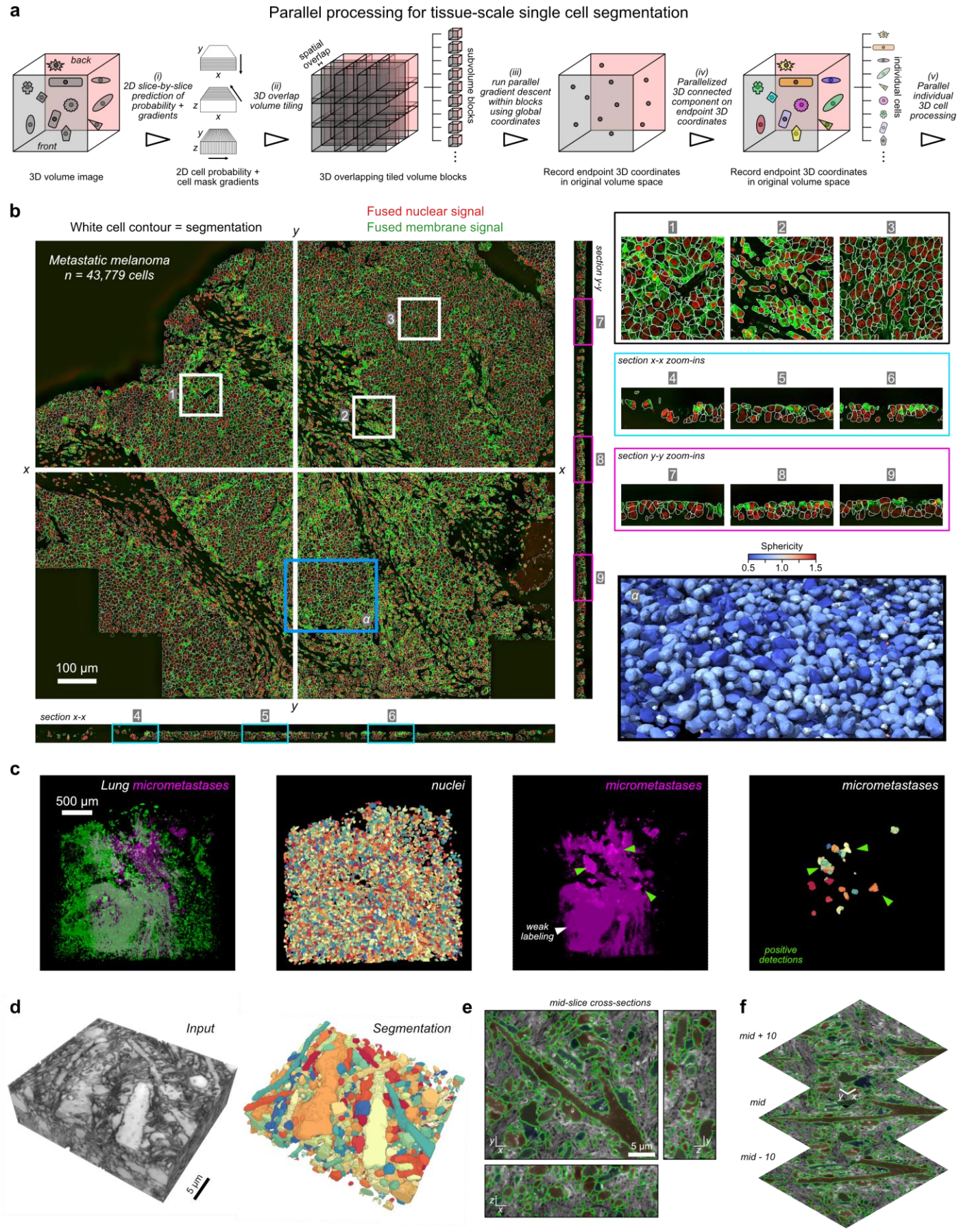
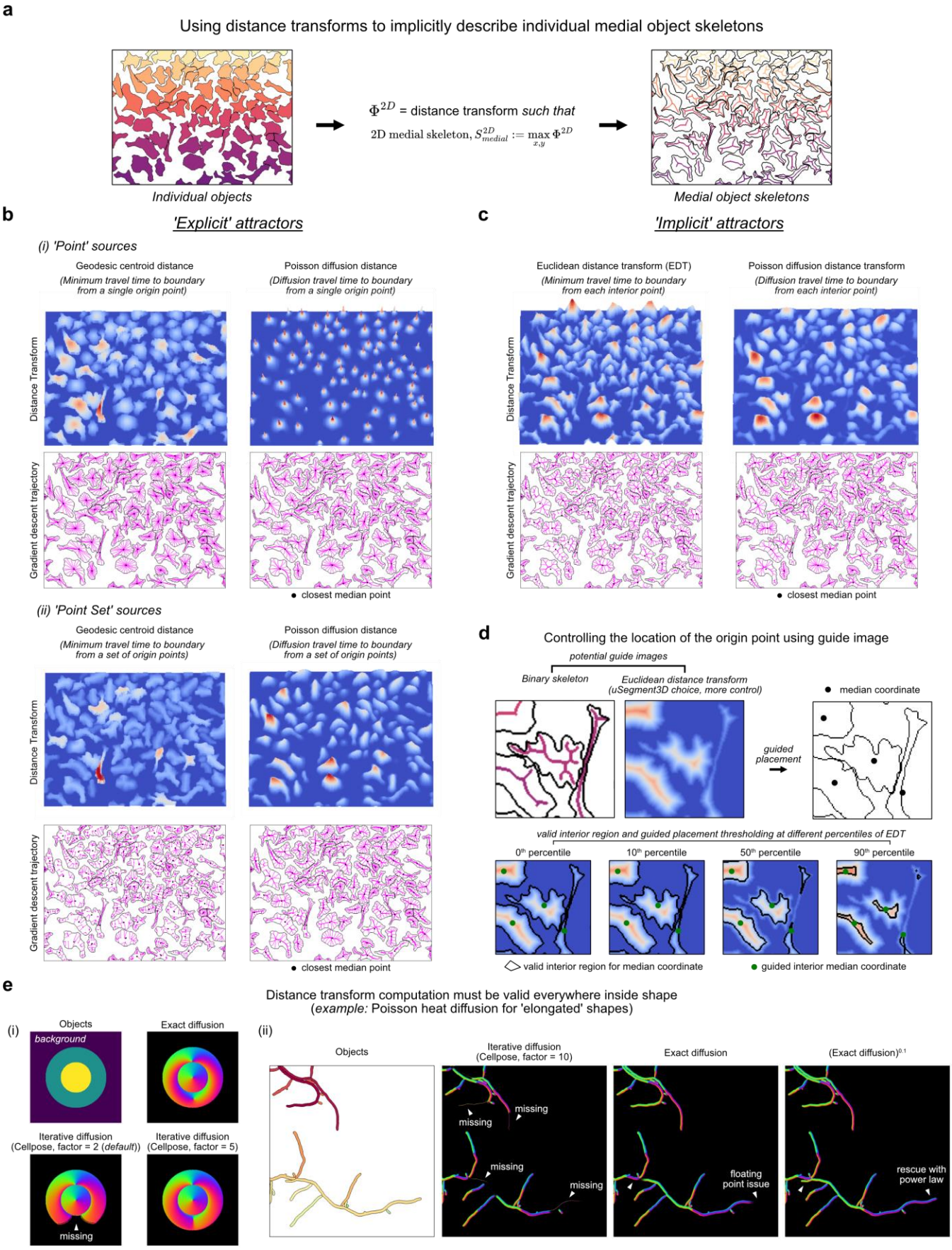


Figure 6. u-Segment3D uses parallel computing for tissue-scale segmentation. a) Schematic of the parallelized gradient descent tracking in overlapped subvolume tiles used by u-Segment3D to facilitate single cell 3D segmentation in tissue. **b)** x-y, x-z, y-z midslice cross-sections of the fused nuclear (red) and membrane (green) signal channels from multiple biomarkers (Methods) for a CyCIF multiplexed patient biopsy of metastatic melanoma with white boundaries to delineate the individual cells in each view (left). Zoom-ins of 3 subregions in x-y (black box, regions 1-3), x-z (cyan box, regions 4-6), y-z (magenta box, 7-9)

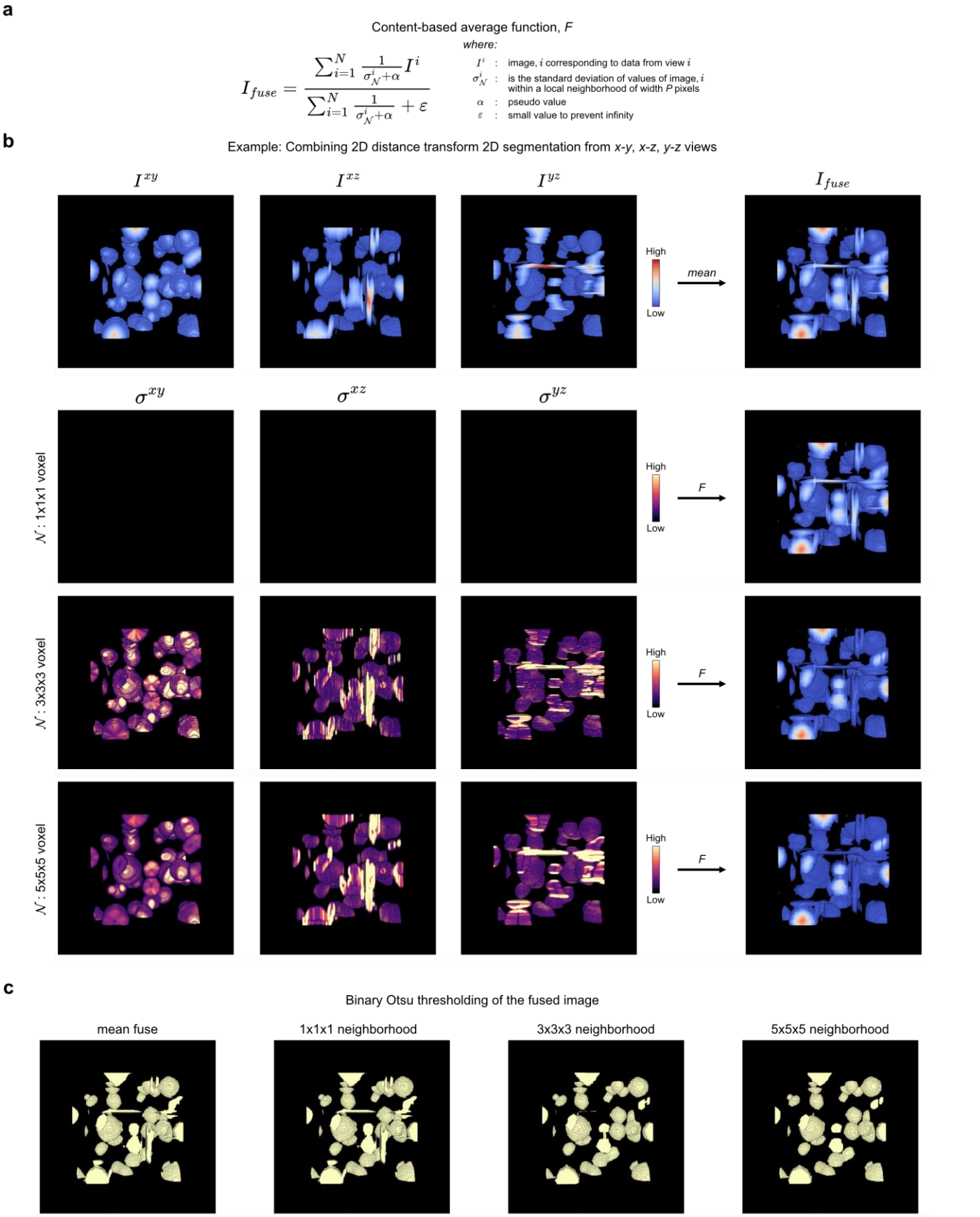
cross-sections (top right). Zoom-in of the extracted 3D cell meshes within the blue rectangle (subregion α) of xy view, heatmap colored by sphericity (sphere = 1). **c)** 3D Segmentation of individual lung nuclei (green) and cancer micrometases (magenta) in cleared tissue. Left-to-right: merged input volume image, individual segmented nuclei from nuclei channel, micrometastases only image showing weak, non-specific staining (white arrow) compared to specific positive staining (green arrow), and final u-Segment3D micrometastases 3D segmentation post-filtered by mean cell intensity. **d)** 3D render of the input coCATs volume (left) and u-Segment3D aggregated cellpose2D 3D segmentation of salient tissue architecture (right). **e)** Mid-slice cross-sections in x-y, x-z, y-z with individual segmentation boundaries outlined in green and its area individual color overlayed with the input image. **f)** Mid $\pm$ 10 z-slice x-y cross-section with individual segmentation boundaries outlined in green and its area individual color overlayed with the input image.

Supplementary Figures



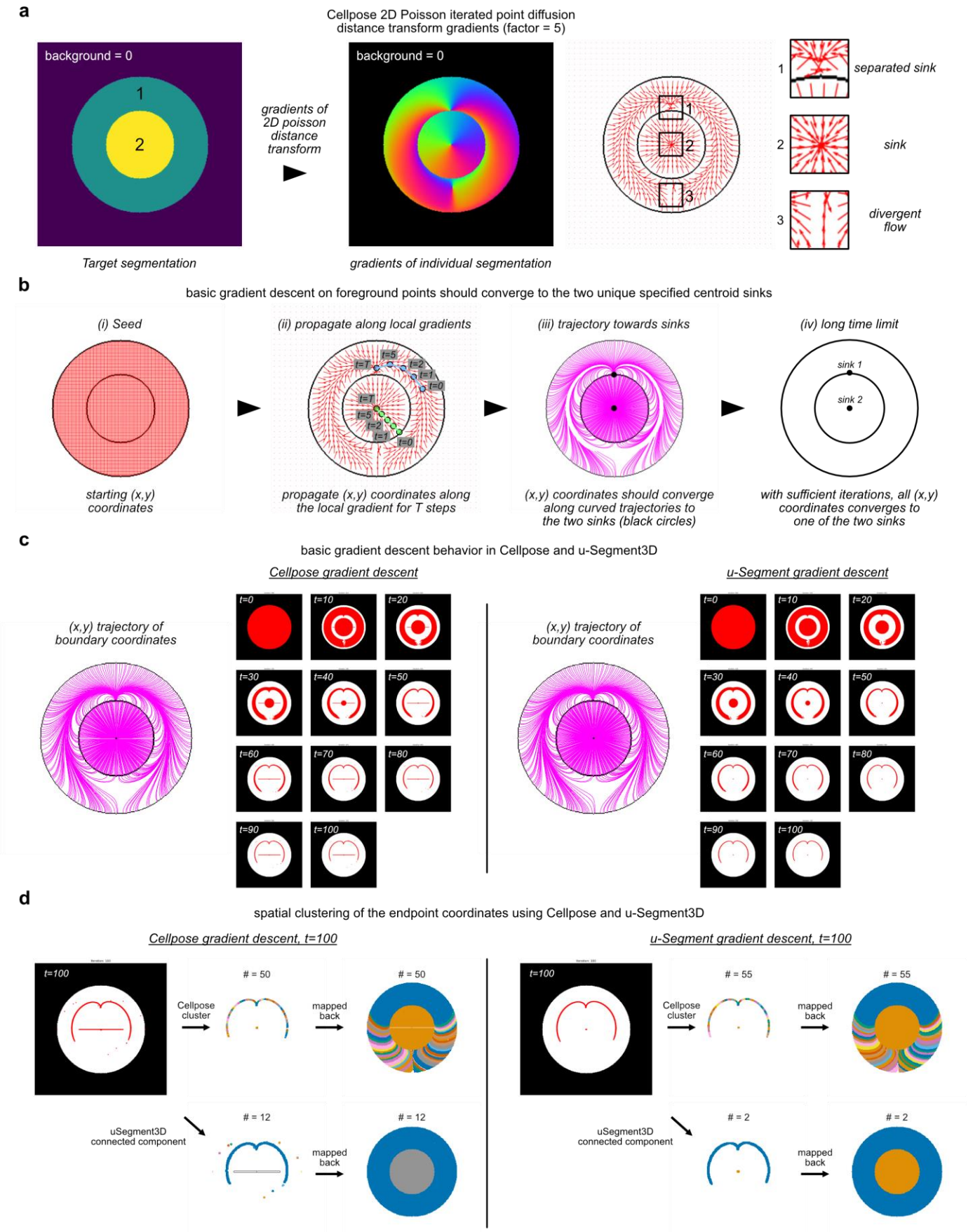
Extended Data Figure 1. 2D distance transforms for 2D-to-3D segmentation in u-Segment3D. a) Schematic illustration of the necessary property of 2D distance transforms for 3D segmentation should have maxima that sample the 2D medial skeleton of individual shapes. **b)** Example of the first- (geodesic, left) and second-order (diffusion, right) shortest distance 'explicit' attractors distance transforms implemented in u-Segment3D where the limit of performing gradient descent is explicitly specified as either (i) a single point-

source on the 2D medial skeleton or (ii) a set of points as the source along the 2D medial skeleton. **c)** Example of the first- (geodesic, left) and second-order (diffusion, right) shortest distance 'implicit' skeletal-based attractor distance transforms implemented in u-Segment3D where performing gradient descent may not necessarily converge stably to the limit. For each example in **b), c)**, distance transform is represented as a relative 3D height map, colored blue (lowest) to red (highest) (top) and trajectory (magenta) of equi-sampled boundary points under gradient descent (bottom). Black point = closest internal shape point to the median shape coordinate. **d)** Illustration of using percentile-based thresholding of the Euclidean distance transform of individual cells as a soft constraint to find the medial centroid 2D coordinate for convex and concave shapes used in computing the 2D point-source distance transforms in c). **e)** The computed unit-normalized 2D gradients for (i) circle in doughnut synthetic shape and (ii) elongated touching bacterial shapes using the iterative simulated diffusion in Cellpose compared to the exact diffusion with boundary conditions solution obtained by LU factorization (Methods) in u-Segment3D. 2D gradients are colored according to direction. The number of simulated diffusion steps in Cellpose equals the factor multiplied by the number of pixels occupied per shape.



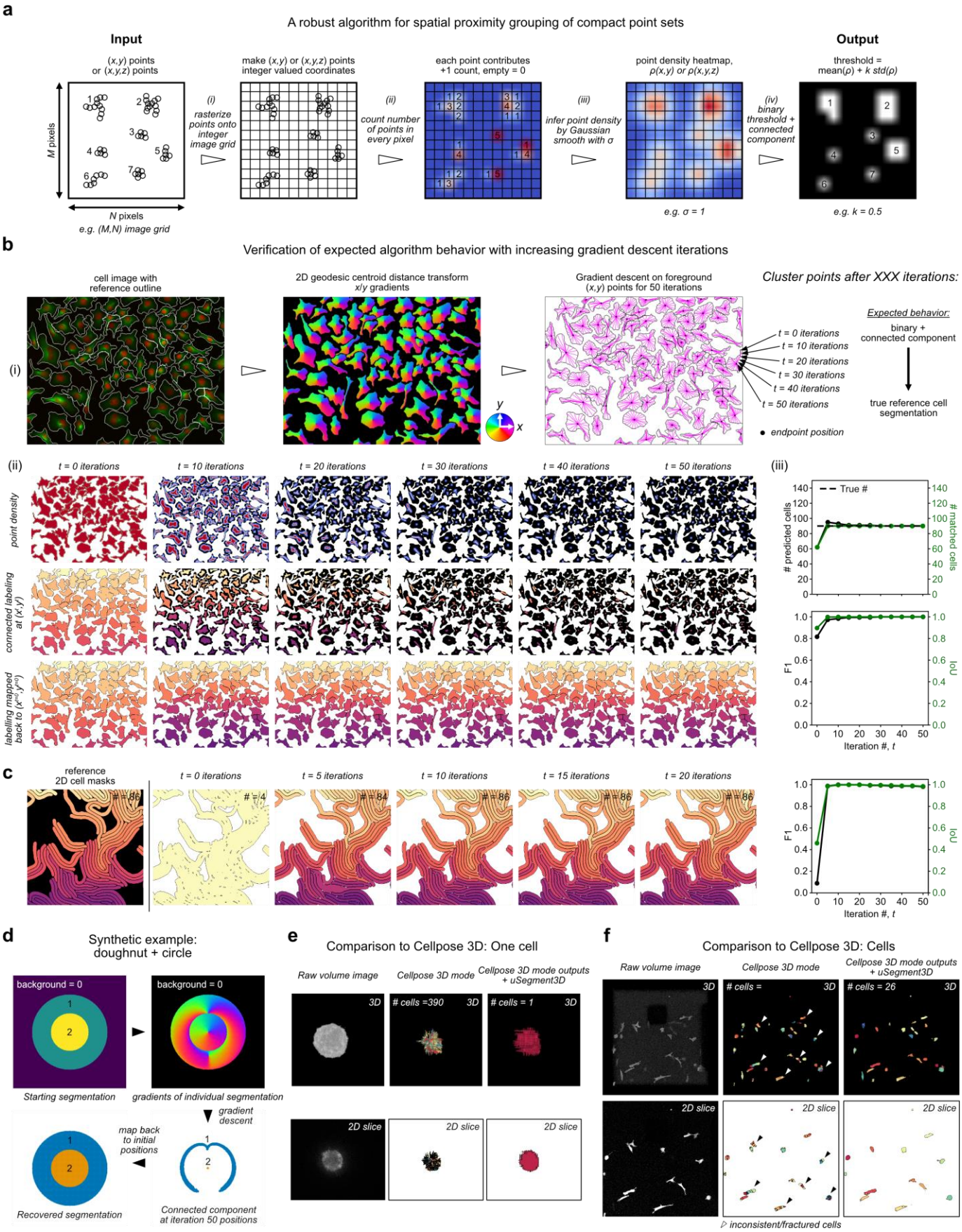
Extended Data Figure 2. Content-based average function for combining data from multiple views for 2D-to-3D segmentation. a) Mathematical definition of the content-based average function as the inverse local variance weighted mean of input image values in an isotropic neighborhood of width *P* pixels. **b)** Example of applying the average function defined by a) to fuse the stacked 2D distance transforms after 2D slice-by-slice segmentation of a volume of spherical cells from xy, xz and yz views. 1st row: Left-to-right, the Euclidean distance transform colored blue (low) to red (high) from the three orthoviews and fused distance transform from pixelwise mean. 2nd row: Left-to-right, the per-pixel local variance weight image σ for each orthoview for a neighborhood of width P = 1 pixel and resultant fused distance transform using the content-

based averaging. 3rd row: Left-to-right, the per-pixel local variance weight image σ for each orthoview for a neighborhood of width P = 3 pixel and resultant fused distance transform using the content-based averaging. 4th row: Left-to-right, the per-pixel local variance weight image σ for each orthoview for a neighborhood of width P = 5 pixel and resultant fused distance transform using the content-based averaging. **c)** Result of applying binary Otsu thresholding on the fused distance transform based on pixelwise mean, and content-based averaging using neighborhood of width P = 1, 3, 5 pixels from left-to-right.



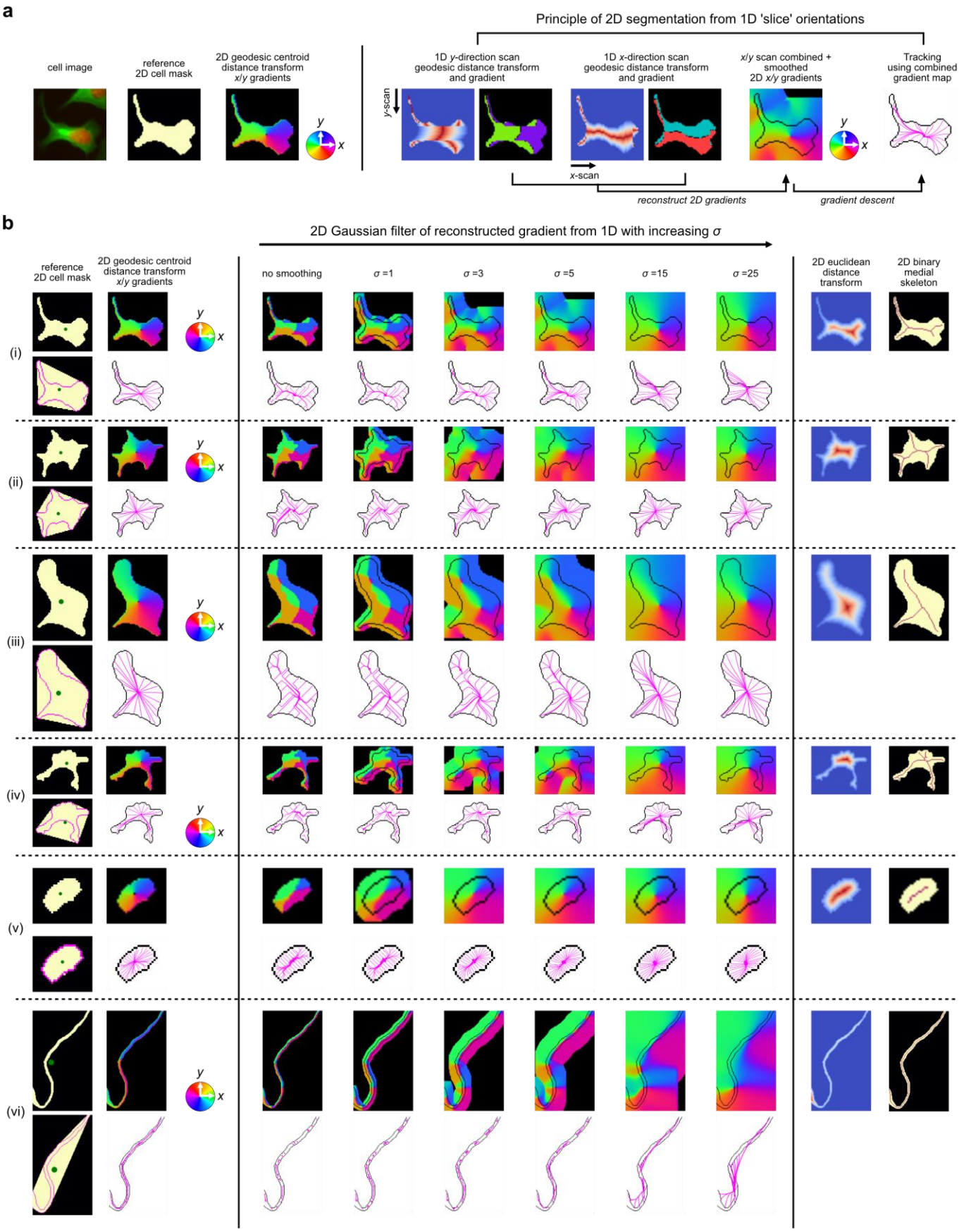
Extended Data Figure 3. Illustration of 2D Gradient descent for reversible 2D shape erosion. a) Cellpose computed 2D gradients for a synthetic shape of a doughnut (labelled 1) surrounding a circle (labelled 2) (left) with the 2D gradients visualized as red arrows (right) and zoomed in at three regions with different types of flow behavior. The number of simulated steps equals the factor = 5 multiplied by the number of pixels occupied per shape. **b)** Schematic of the expected behavior when gradient descent is iteratively applied to propagate the initial foreground (x,y) coordinates with the 2D gradients with the limit being convergence to 2 black centroids. **c)** Observed point trajectories (magnenta lines) and snapshots of coordinate positions (red points in images) running gradient descent in cellpose (left) vs u-Segment3D (right)

for 100 iterations. **d)** Recovered cell shapes based on applying Cellpose (top) or u-Segment3D (bottom) spatial proximity clustering on the final coordinate positions after 100 iterations of Cellpose (left) or u-Segment3D (right) gradient descent.



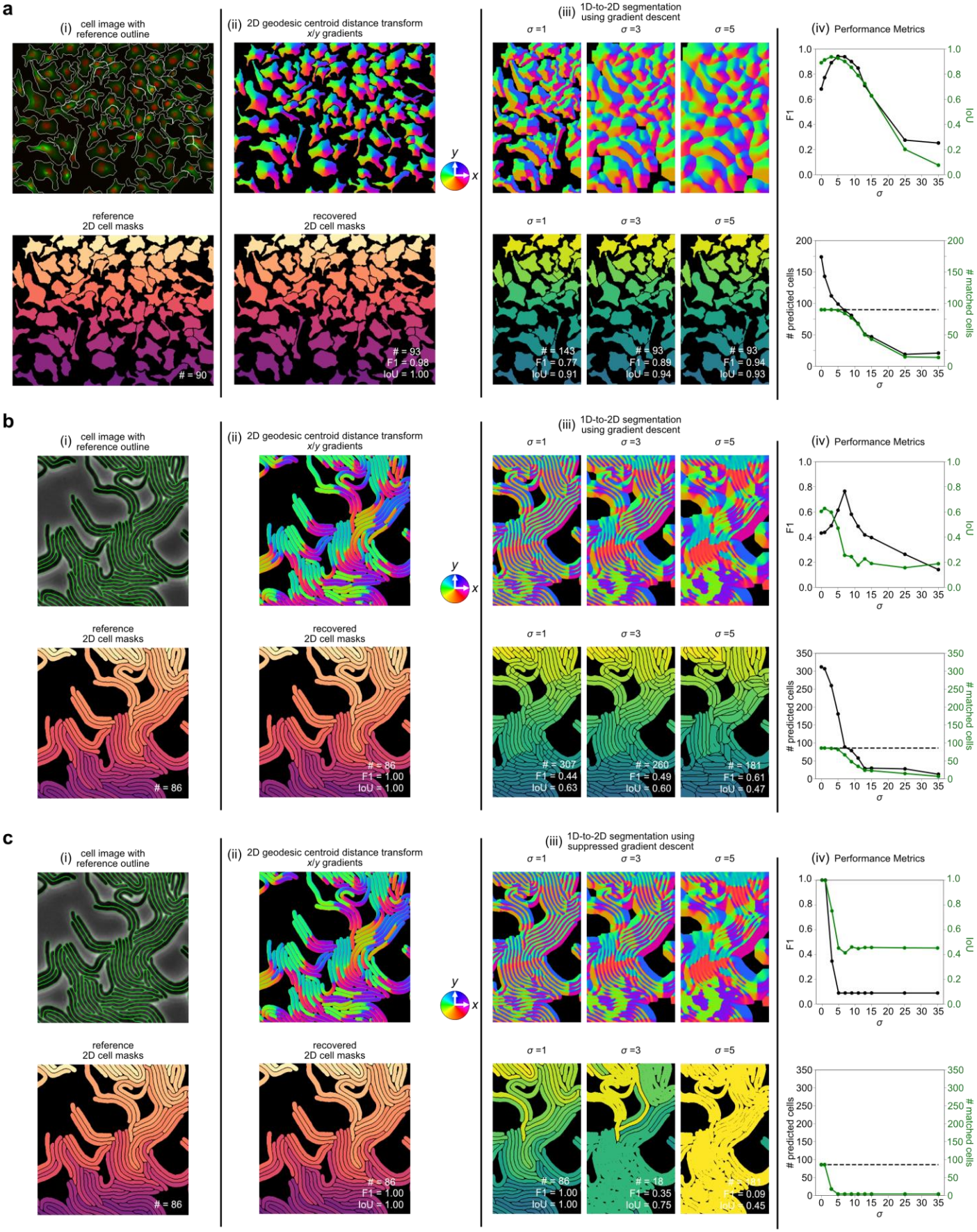
Extended Data Figure 4. Using image-based spatial connected components to robustly identify distinct spatially compact point sets. a) 2D Illustration of the image-based connected components spatial clustering approach in u-Segment3D involving left-to-right, rasterization of floating-point coordinates onto a discrete image pixel grid, building a count of the number of points in each pixel, approximate Gaussian kernel density estimation using a Gaussian filter of σ, binary thresholding on the mean density and subsequent connected component analysis to identify distinct spatial clusters. **b)** Verification of stable algorithm behavior by application to foreground (x,y) coordinates after propagation of XXX iterations with gradient descent. 2D gradients are computed using the 2D geodesic centroidal point distance transform. (i) Schematic of the

experiment setup for an example 2D cell segmentation with densely touching cells (top panel). (ii) The point density, connected component labelling at current (x,y) coordinates and labeling mapped back to initial (x,y) coordinates (top-to-bottom) after t=0, 10, 20, 30, 40, 50 iterations of gradient descent (left-to-right). (iii) Plot of number (#) of distinct predicted cells (left, black-colored y-axis and line) and # of matched cells with reference cell segmentation (right, green-colored y-axis and line) with iteration #, *t* (top). Dashed black horizontal line indicates the true cell number. Plot of F1 score of matching with reference cells (left, black-colored y-axis and line) and the mean intersection-of-union (IoU) of matched cells with reference (right, green-colored y-axis and line) with iteration #, *t* (bottom). **c)** Reference 2D cell segmentation of elongated touching bacteria (left), identified unique cells by spatial connected component at gradient descent propagated coordinates after t = 0, 5, 10, 15, 20 iterations (middle), and Plot of F1 score of matching with reference cells (left, black-colored y-axis and line) and the mean intersection-of-union (IoU) of matched cells with reference (right, green-colored y-axis and line) with iteration #, *t*. **d)** Image-based connected component applied to recover a doughnut (region 1) surrounding a circle (region 2) after 50 iterations of gradient descent. **e)** Comparison of using image-based connected component to construct robust segmentations from Cellpose 3D mode outputs for an isolated noisy single cell from the 3D cell tracking challenge. Left-to-right: 3D render of raw volume, Cellpose 3D mode segmentation with diameter=50 and cellprob_threshold=0, reparsed 3D segmentation using u-Segment3D gradient descent and connected component analysis (top row) and corresponding mid x-y slice (bottom row). **f)** Comparison of using image-based connected component to construct robust segmentations from Cellpose 3D mode outputs for a noisy image of multiple cells of elongated morphologies from the 3D cell tracking challenge. Left-to-right: 3D render of raw volume, Cellpose 3D mode segmentation with diameter=15 and cellprob_threshold=-1.2, reparsed 3D segmentation using u-Segment3D gradient descent and connected component analysis (top row) and corresponding mid x-y slice (bottom row). Black arrowheads highlight examples of fractured single cells due to unstable spatial clustering in Cellpose.



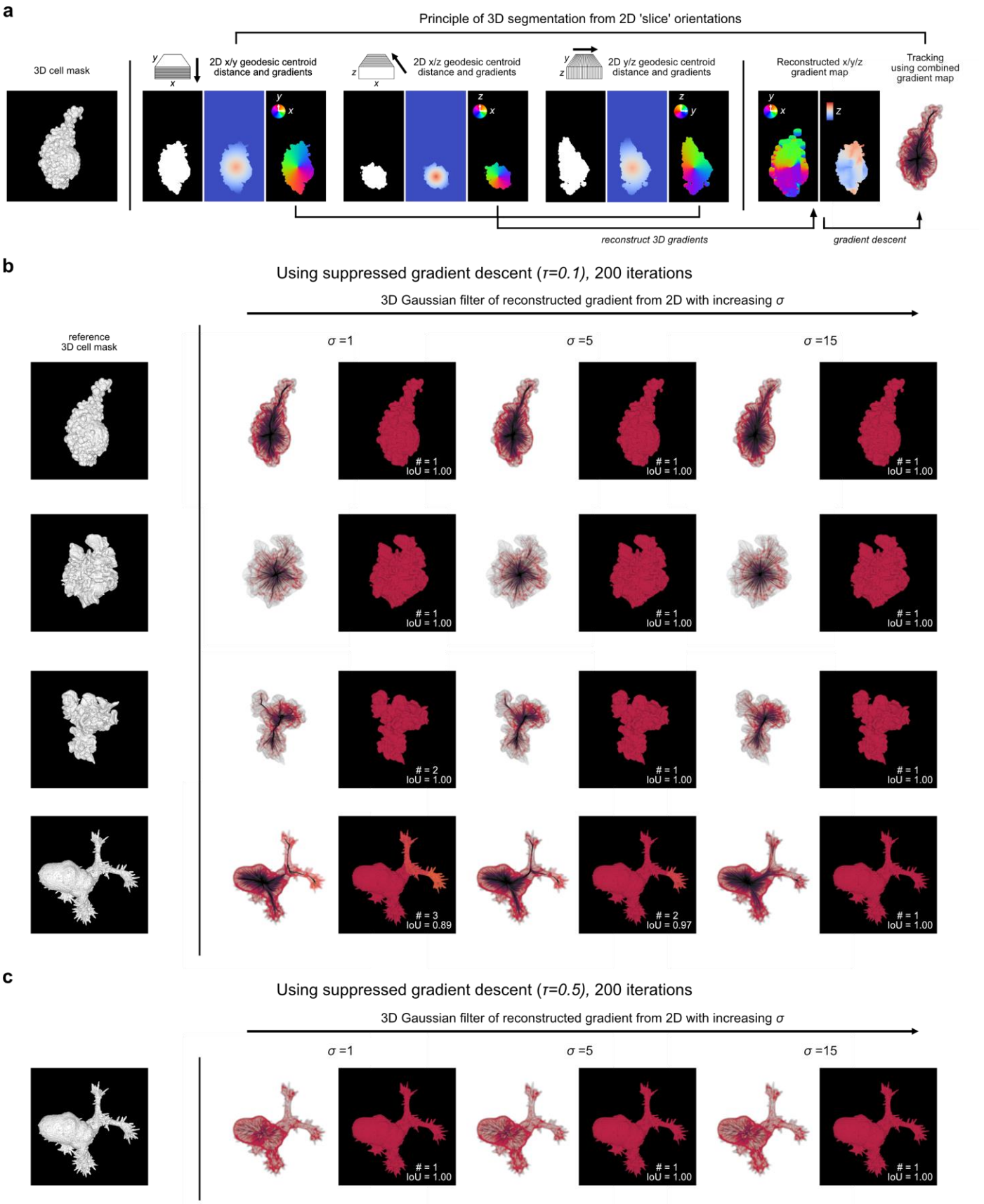
Extended Data Figure 5. 1D-to-2D segmentation for single cells from synthetic ideal 1D segmentation stacks. a) Schematic of the workflow to investigate 1D-to-2D segmentation. Reference cell segmentation and ideal 2D geodesic distance transform to reconstruct (left). Reconstruction of 2D gradient field from computed 1D gradients (right). Left-to-right: Determination of the y-direction and x-direction gradient by computing the distance transform of each pixel in 1D slices to the respective slice centroid and taking the gradient; combining the x- and y- gradient into a 2D gradient field and smoothing with a Gaussian σ; performing gradient descent on the smoothened 2D gradients to propagate all interior points to a unique centroid (magenta line trajectories). **b)** Gradient descent behavior using Gaussian filter of increasing σ on the

initial reconstructed 2D gradients for individual cell examples representing the diversity of morphologies. Example of approximately star-convex shapes: (i) Montage of 4 images depicting reference cell shape and its convex hull image with green point representing their centroid coordinate and the cell's exact 2D geodesic centroid distance transform with associated gradient descent trajectory to reconstruct (left panels). Observed gradient descent trajectory (magenta line) when reconstructed 2D gradients is isotropically smoothed with Gaussian filter of increasing σ (middle panels). Comparison of the gradient descent trajectory with the implicit medial skeleton specified by the 2D Euclidean distance transform and explicit medial skeleton from morphological operations (right panels). (ii), (iii) are further examples of approximate star-convex shapes as in (i). (iv) represents a branching shape whose branch lengths are comparable to the cell body, (v) a convex shape, and (vi) a thin vessel-like shape with panels depicted as for (i).



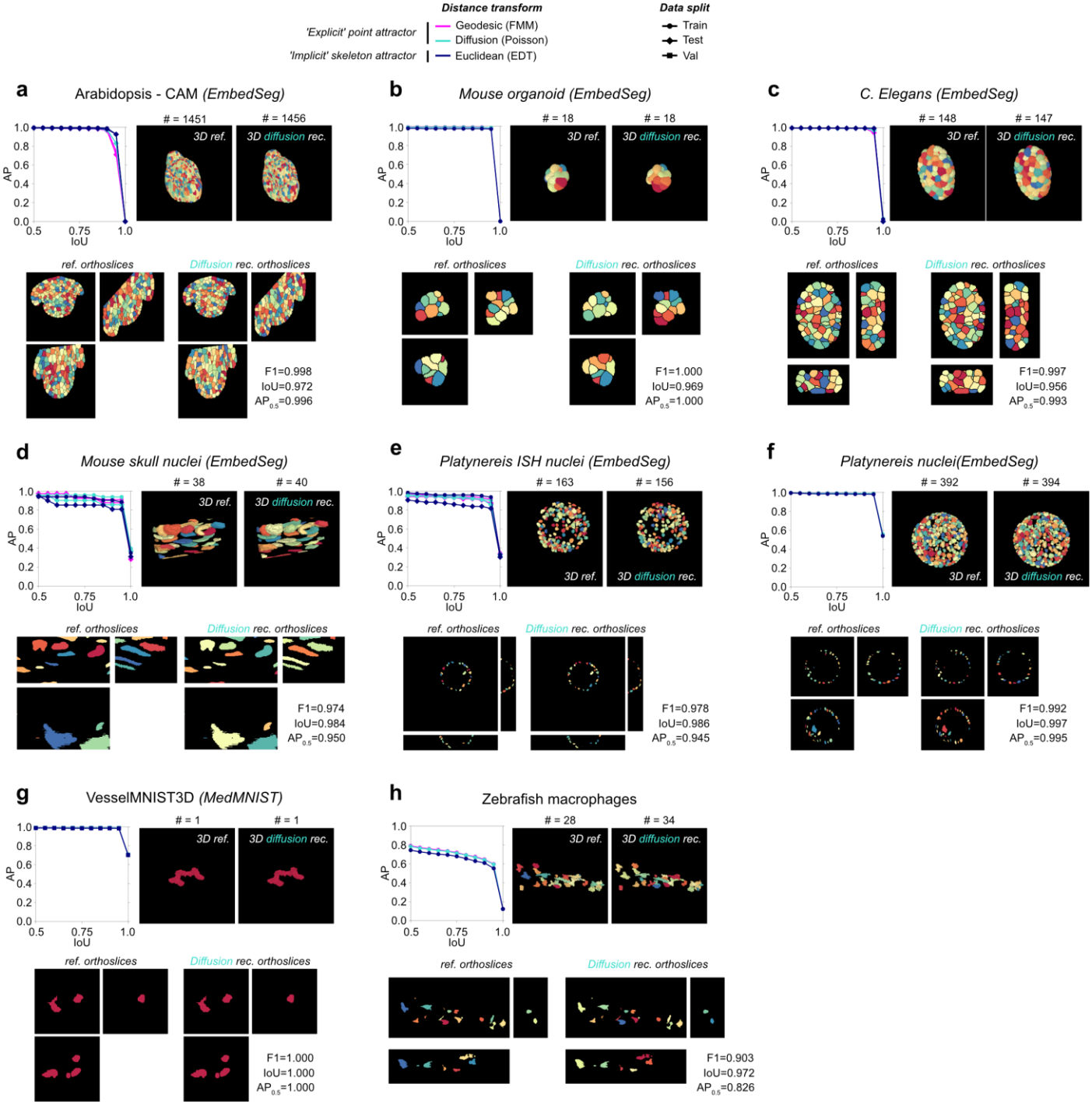
Extended Data Figure 6. 1D-to-2D segmentation for dense clustered cells from synthetic ideal 1D segmentation stacks. a) Representative example of a dense 2D cell culture with diverse morphologies with (i) reference cell boundaries overlaid and delineated in white (top) and uniquely colored segmented masks (bottom). (ii) Exact unit-normalized 2D gradients geodesic centroid distance transform colored by direction (top) and the recovered cell masks using connected component analysis after 100 iterations of gradient descent with step-size one pixel. (iii) Reconstructed 2D gradient from 1D after Gaussian filtering with increasing σ left-to-right (top) and corresponding recovered cell masks using connected component analysis

after 100 iterations of gradient descent. (iv) Recovery performance of the 1D-to-2D segmentation with increasing σ. Plot of F1 score of matching with reference cells (left, black-colored y-axis and line) and the mean intersection-of-union (IoU) of matched cells with reference (right, green-colored y-axis and line) (top). Plot of the number (#) of distinct predicted cells (left, black-colored y-axis and line) and # of those that could be matched with reference cells (right, green-colored y-axis and line) (bottom). Dashed black horizontal line indicates the true cell number. Evaluation was performed after 100 iterations of gradient descent. **b)** Representative example of a dense 2D cell culture where each cell has highly elongated, vessel-like morphology. Panels (i)-(iv) as in **a)**. **c)** Same example image and panels (i)-(iv) as in **b)** but using suppressed gradient descent where the step-size $\eta = \frac{1}{1+\tau \cdot t}$, is attenuated with gradient descent iteration number, t and $\tau = 1$ (Methods).

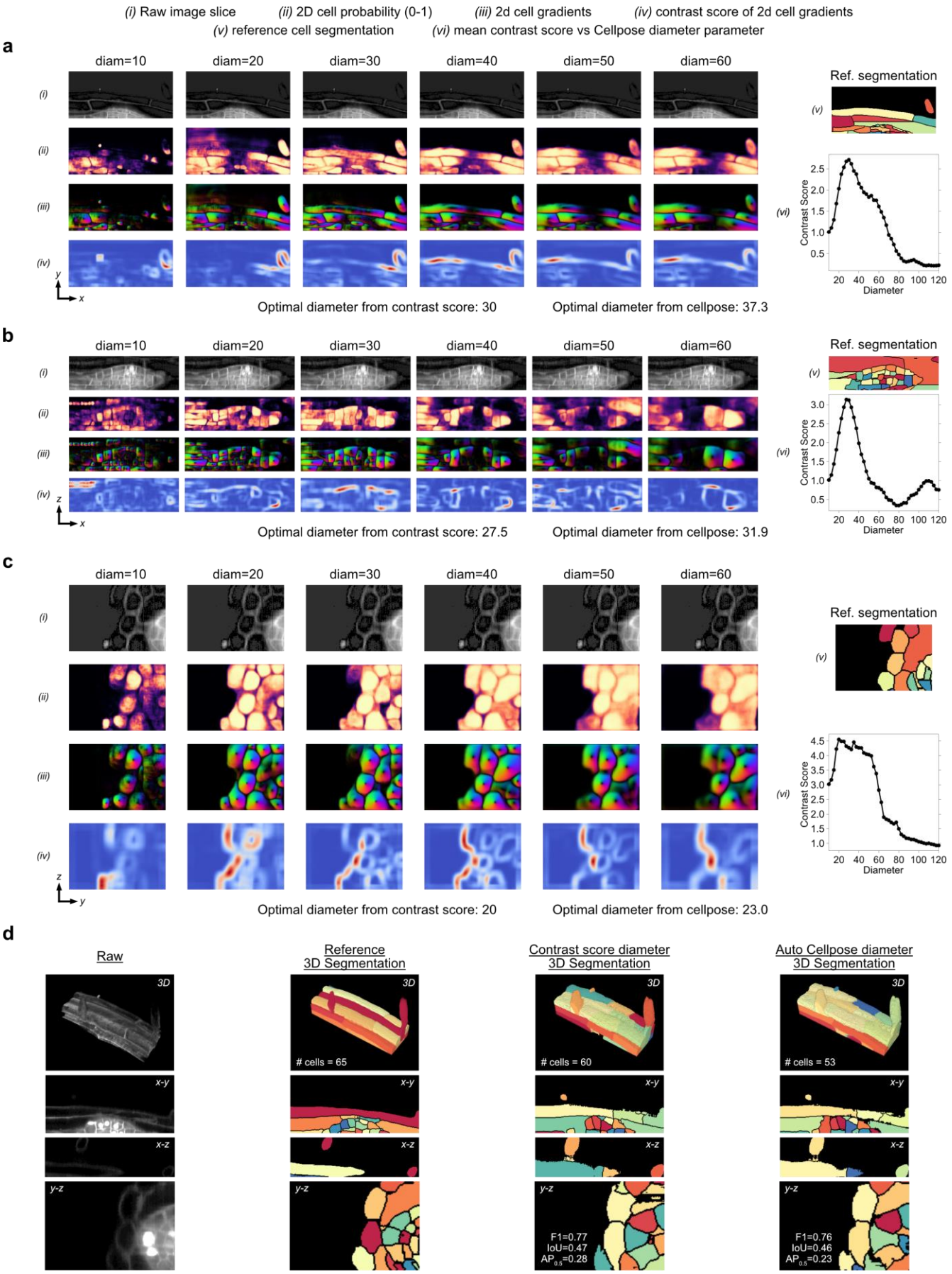


Extended Data Figure 7. 2D-to-3D segmentation reconstruction for single cells from synthetic ideal 2D segmentation stacks. a) Illustration of the reconstruction experiment given a single 3D cell segmentation (left), by generating 2D gradients slice-by-slice in xy, xz, yz views, treating each disconnected spatial component as a unique 2D cell (middle) and performing gradient descent on the reconstructed 3D xyz gradient followed by connected component analysis on the final advected 3D coordinates (right). **b)** Reconstruction examples for cells with blebs (1st row), lamellipodia (2nd, 3rd rows) and filopodia (4th row). In each row, left-to-right: reference binary 3D cell segmentation, the 3D gradient descent trajectory (left) and reconstructed 3D segmentation (right) for Gaussian filtering of the 3D reconstructed gradients with $\sigma = 1,5,15$,

using suppressed gradient descent ($\tau = 0.1$) with momentum (0.95). **c).** Reconstruction example of the same cells with filopodia in b) for post- Gaussian filtering with $\sigma = 1,5,15$ (left-to-right), using suppressed gradient descent with greater decay ($\tau = 0.5$) with momentum (0.95).

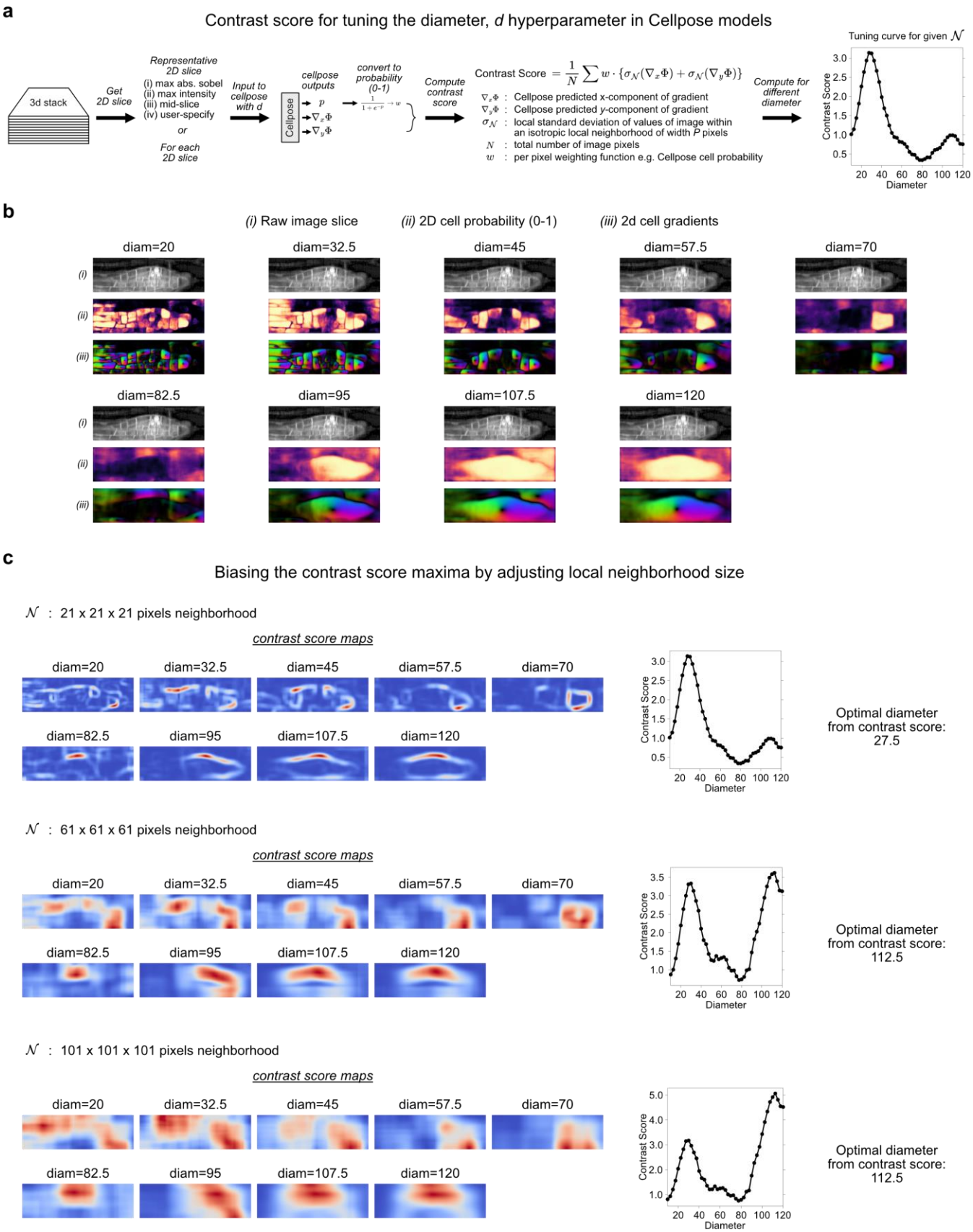


Extended Data Figure 8. Reconstruction performance of 3D cell segmentation from synthetic ideal 2D segmentation stacks for real datasets. Reconstruction performance measured by the mean average precision curve (Methods) using three different 2D distance transforms for all datasets not included in Fig. 2d-f. **a**) Arabidopsis-CAM, **b)** mouse organoid, **c)** *C.Elegans*, **d)** mouse skull nuclei, **e)** *Platynereis* ISH nuclei, **f)** *Platynereis* nuclei, **g)** vesselMNIST3D and **h)** zebrafish macrophages. For each dataset, top row, left-to-right: average precision vs intersection over union (IoU) curve; 3D rendering of reference segmentation, point-based centroid diffusion distance transform reconstructed 3D cell segmentation. Bottom row, left-to-right, the respective midplane orthoslices in the three orthogonal views. All available data splits is used for each dataset except VesselMNIST3D which we use only the validation split (Methods). See Suppl. Table 1 for number of objects and images in each split.

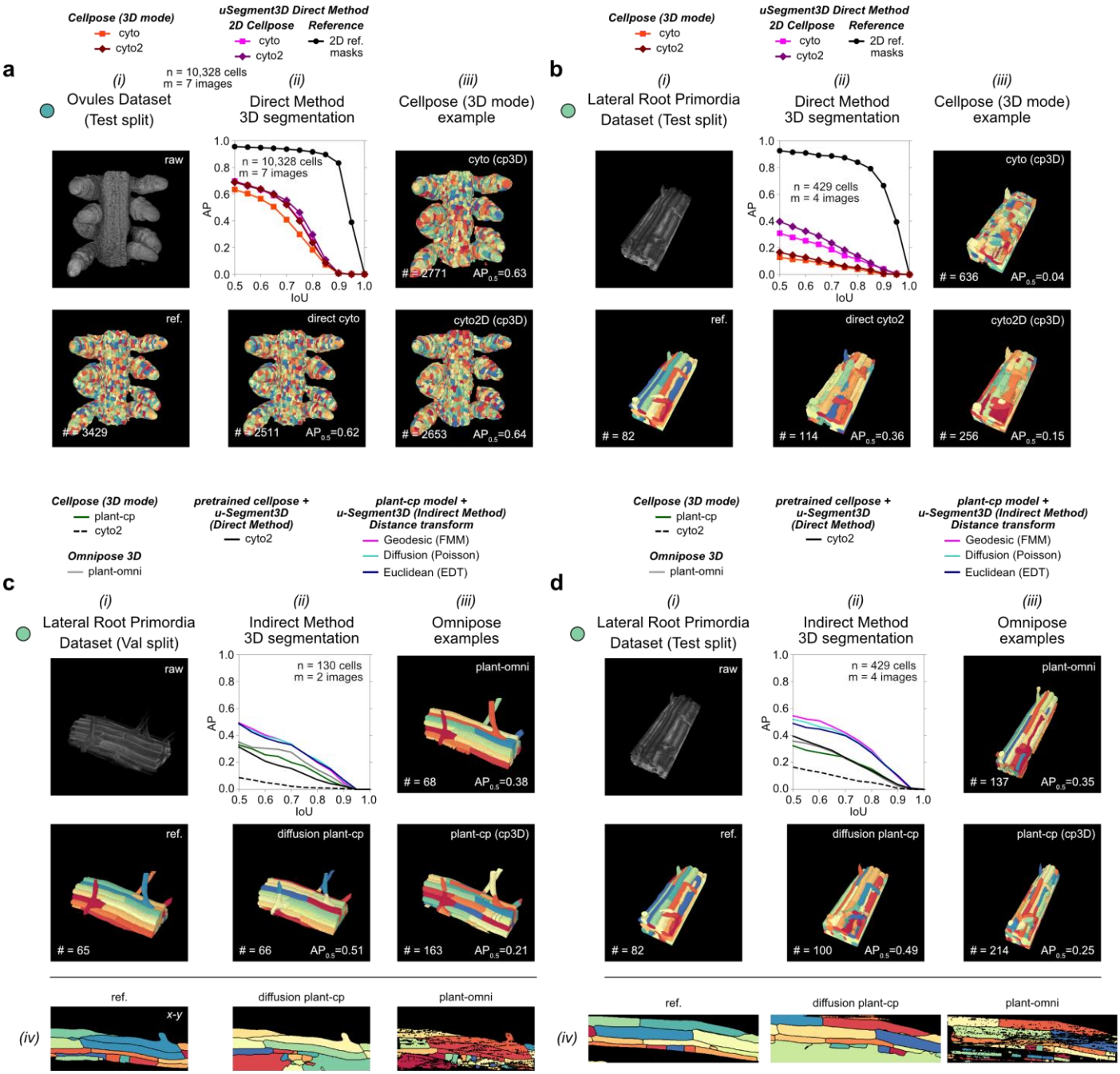


Extended Data Figure 9. The diameter parameter in pretrained Cellpose models should be individually set in orthogonal views. a) Cellpose 'cyto2' model outputs and per-pixel u-Segment3D contrast score on the most in-focus (Methods) 2D xy-slice of a Lateral Primordia as diameter (diam) is increased. (i) Raw input image slice, the same for all values of diameter. (ii) Normalized (0-1) Cellpose 2D pixel probability map colored black=0 to yellow=1. (iii) Unit-normalized Cellpose 2D predicted gradients colored by direction. (iv) Contrast score of predicted cellpose 2D gradient (Methods, Extended Data Figure 10a). (v) Corresponding reference cell segmentation for the raw 2D input. (vi) Mean contrast score averaged over the image for each

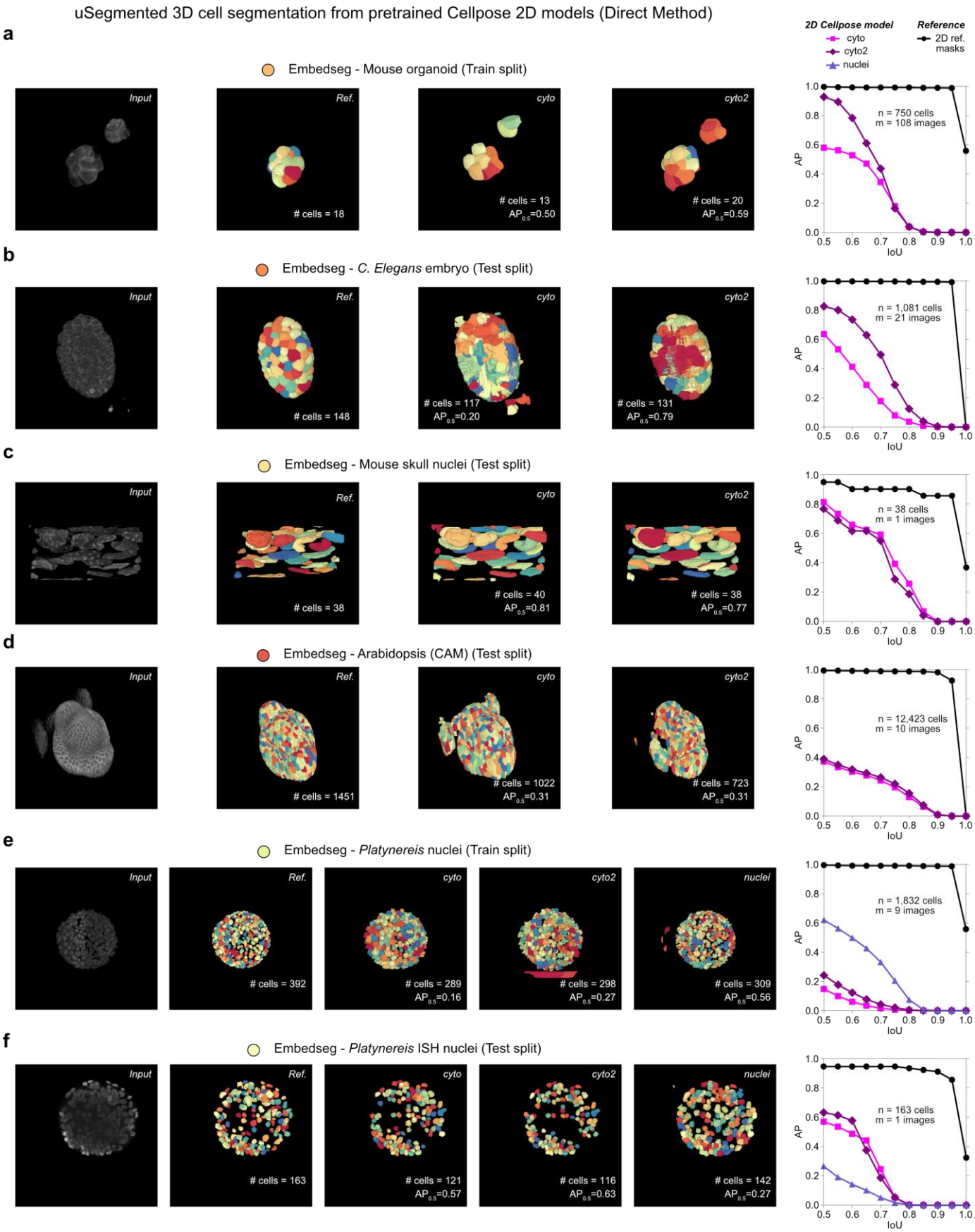
value of diam. The diameter with maximum contrast score is taken by u-Segment3D as the optimal diameter. **b)** Same as in **a)** for the most in-focus 2D xz-slice and **c)** the most in-focus 2D yz-slice. **d)** From left-to-right, 3D render of the raw 3D volume, reference 3D segmentation, u-Segment3D consensus 3D segmentation using the direct method and diameter in each orthoview set by contrast score or by the Cellpose ‘cyto2’ model (top), and corresponding mid-plane orthoslices in all three views (below).



Extended Data Figure 10. Semi-automatic determination of the diameter parameter in pretrained Cellpose models using local variance. a) Schematic illustration of the computation and definition of a contrast score function, evaluated based on a user-specified local neighborhood of width P pixels to evaluate Cellpose outputs for a 2D image when the diameter parameter is set to d. **b)** (i) Raw input image slice, (ii) **n**ormalized (0-1) Cellpose 2D pixel probability map colored black=0 to yellow=1, and (iii) unit-normalized Cellpose 2D predicted gradients colored by direction for 9 equisampled diameters in the range $d = [20, 120]$. **c)** Contrast score maps (colored blue-to-red for low-to-high values) for the same d as in b) (left) and resulting contrast score function and optimal diameter inferred (right), given a specified neighbourhood of width $P = 21, 61, 101$ pixels (top-to-bottom).

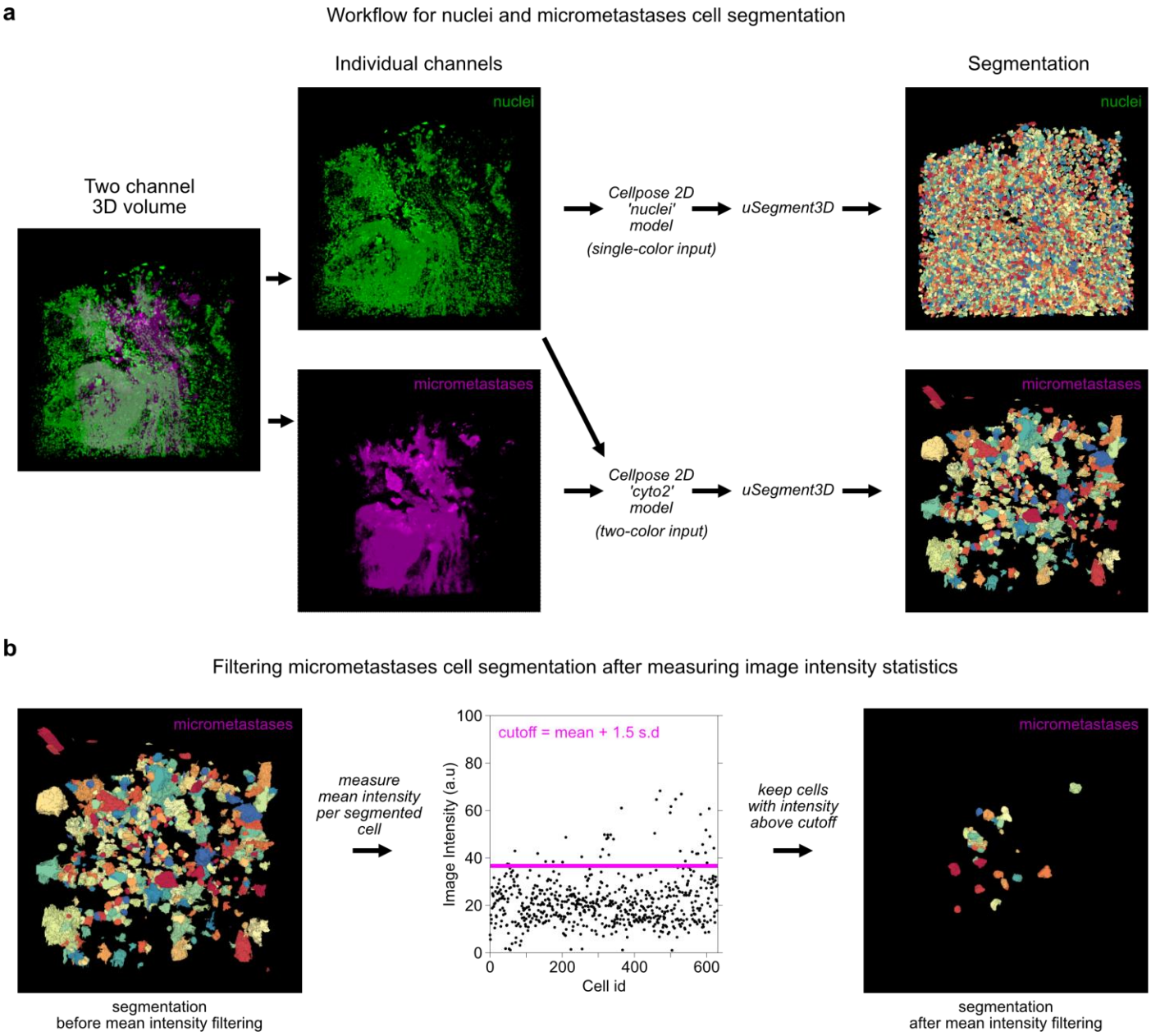


Extended Data Figure 11. Performance of u-Segment3D using pretrained vs specialized plant Cellpose segmentation models. a) Performance of pretrained Cellpose models with u-Segment3D vs Cellpose 3D mode on test split of Ovules. (i) example volume and corresponding reference segmentation; (ii) AP curves of all models, with best 3D segmentation reconstructed from ideal 2D slices (black line) (top) and segmentation of best pretrained model with u-Segment3D (bottom); (iii) Cellpose 3D mode segmentations using cyto (top) or cyto2 (bottom) models. **b)** Performance of pretrained Cellpose models with u-Segment3D vs Cellpose 3D mode on test split of Lateral Root Primordia (LRP) dataset. (i)-(iii) similar to a). **c)** Performance on val split using pretrained Cellpose models with u-Segment3D or Cellpose 3D mode, plant-cp: a specialized Cellpose 2D model trained on LRP with u-Segment3D or Cellpose 3D mode, and plant-omni: a specialized Omnipose 3D model trained on LRP natively in 3D. (i)-(ii) similar to a). (iii) Native 3D segmentation using plant-omni (top) or 2D-to-3D segmentation using Cellpose 3D mode with plant-cp (bottom). (iv) mid xy slice of the reference, u-Segment3D diffusion centroid transform aggregated plant-cp and 3D plant-omni segmentation (left-to-right). **d)** Same as c) for the test split of LRP.



Extended Data Figure 12. Performance of 2D-to-3D segmentation for real datasets using u-Segment3D and pretrained cellpose2D model outputs. a) 3D cell segmentation performance of the mouse organoid (in the Embedseg paper) for the train data split, n=740 cells, m=108 volumes) using pretrained Cellpose 2D models with u-Segment3D and the direct method illustrated in Fig.3a. Left-to-right: 3D rendering of the raw volume, reference 3D segmentation, generated 3D segmentations for each Cellpose 2D model and the combined average precision (AP) curve coplotted with the AP curve of the best reconstruction with synthetic reference 2D segmentations in the three orthoviews (black line with circles). The same as **a)** for **b)** C. Elegans embryo (test data split, n =1,081 cells, m = 21 images), **c)** mouse skull nuclei (test data split,

n=38 cells, m=1 image), **d)** Arabidopsis (CAM) (test data split, n=12,424 cells, m=10 images), **e)** Platynereis nuclei (train data split, n=1,832 cells, m=9 images), **f)** Platynereis ISH nuclei (test data split, n=163 cells, m=1 image). For the Platynereis nuclei which are approximately spherical, we additionally evaluated the performance of the Cellpose ‘nuclei’ 2D model (light purple line with triangles).



Extended Data Figure 13: u-Segment3d generated consensus 3D segmentations can be filtered with image statistics to improve specificity for weakly labeled cells. a) Schematic of the workflow and 3D rendering of the resulting segmentation using the nuclei only stained channel (green) to segment all lung nuclei and using both nuclei (green) and micrometastases stain (magenta) channels to segment micrometastases with Cellpose and u-Segment3D. **b)** Schematic of the procedure to leverage the micrometastases stain to post-filter out mis-segmented micrometases (left) by measuring the mean intensity in each segmented cell (middle) and performing a global statistically determined cutoff, keeping the subset of cell segmentations with mean intensity above the cutoff (right).

Supplementary Tables

Supplementary Table 1. Summary of datasets used for validation of u-Segment3D. Unless otherwise noted, all images were first resampled to isotropic voxel resolution by appropriate downsampling of xy slices and then resized isotropically by the indicated rescaling factor.

Plants

Name	Description	Pixel Size (Z,Y,X) [μm^3]	Used Microscope	Isotropic Rescaling factor	Total #images (Train/Test/Val)	Total #cells
Lateral Primordia[8]	*Arabidopsis thaliana* lateral root of the line sC111 were used at 5 day post germination	(0.25, 0.1625,0.1625)	Multi-view Selective Plane Illumination Microscopy	0.5	21 / 4 / 2	2,068 / 429 / 130
Ovules[8,132]	*Arabidopsis thaliana ovules* stained with SR2200 and TO-PRO-3 iodide	(0.235, 0.075, 0.075) *	Confocal laser scanning microscopy	1	22 / 7 / 2	23,860 / 10,328 / 2,839

* We find (0.235, 0.15, 0.15) to be more isotropic and expected appearance.

Lateral primordia was downloaded from https://osf.io/2rszy/

Ovules was downloaded from https://osf.io/w38uf/

Embedseg

Name	Description	Pixel Size (Z,Y,X) [μm^3]	Used Microscope	Isotropic Rescaling factor	# images (Train/Test/Val)	Total # cells
Arabidopsis-Cells-CAM-small[133,134]	*Arabidopsis* Thaliana YFP membrane labelled and imaged between 24 and 28 days after germination	(0.26, 0.22, 0.22)	Confocal Microscopy	1	11 / 10 / -	12,016 / 12,423 / -
C.elegans-Cells-HK[56]	C.elegans embryos membrane labeled.	(0.25,0.25,0.25)	Confocal Microscopy	1	54 / 21 / -	3,479 / 1,081 / -
Mouse-Organoid-Cells-CBG[17]	Mouse Embryonic Stem Cells, R1 cell line, labeled membrane	(1.0, 0.1733, 0.1733)	Selective Plane Illumination Microscopy	0.5	108 / - / -	750 / - / -
Mouse-Skull-Nuclei-CBG[17]	Nuclei of the skull region of developing mouse embryos, labeled with DAPI	(0.200, 0.073, 0.073) *	Inverted Zeiss LSM 880 Microscope	0.5	2 / 1 / -	150 / 38 / -
Platyneris-Nuclei-CBG[17]	Nuclei of whole-mount *Platynereis dumerilli* specimens at stages between 0 to 16 hours post fertilization, injected with a fluorescent nuclear tracer	(2.031, 0.406, 0.406)	Simultaneous Multi-view Light-Sheet Microscopy	1	9 / - / -	1832 / -/ -

Platyneris-ISH-Nuclei-CBG[17]	Nuclei of whole-mount *Platynereis dumerilli* specimens at stage of 16 hours post fertilization, labeled with DAPI	(0.45, 0.45,0.45)	Laser Scanning Confocal Microscopy	1	2 / 1 / -	486 / 159 / -

* For Mouse-Skull-Nuclei-CBG we found that pretrained Cellpose 2D models and u-Segment3D performed better without resizing to isotropic voxels and applying isotropic rescaling factor.

All of these 3D datasets were downloaded from https://github.com/juglab/EmbedSeg/releases/tag/v0.1.0

VesselMNIST3D

Name	**Description**	**Pixel Size (Z,Y,X)**	**Used Microscope**	**Isotropic Rescaling factor**	**# images (Train/Test/Val)**	**Total # cells**
VesselMNIST3D[135,136]	Derived from the IntrA dataset. Vessel segments were generated from 103 3D meshes of entire brain vessels.	(1,1,1)	Time-of-Flight Magnetic Resonance Angiography (TOF-MRA)	9.14	1,335/382/192 (*191)	1,335 / 382 / 191^

^We only used the val split as all images were single-component and morphological properties similar across data splits.

* 1 of the images in the val split was found to be blank.

The dataset was downloaded as .npz from https://zenodo.org/records/10519652.

DeepVesselNet

Name	**Description**	**Pixel Size (Z,Y,X)**	**Used Microscope**	**Isotropic Rescaling factor**	**# images (Train/Test/Val)**	**Total # cells**
DeepVesselNet[10,137,138]	Synthetic generated dataset with images of 325 x 304 x 600. Vessel intensities were randomly chosen in the interval [128, 255] and non-vessel intensities from the interval [0 − 100]. Gaussian noise was then randomly applied.	(1,1,1)	-	1	136 (135*) / - / -	410* / - / -

* We could not download and unzip 1 of the raw images in the archive.

The dataset was downloaded from https://github.com/giesekow/deepvesselnet/wiki/Datasets

Zebrafish macrophages

Name	**Description**	**Pixel Size (Z,Y,X) [μm^3]**	**Used Microscope**	**Isotropic Rescaling factor**	**# images (Train/Test/Val)**	**Total # cells**
Zebrafish-macrophages[83]	Human U-2 OS osteosarcoma cancer cells, labelled with pVimentin-PsmOrange, were injected into zebrafish larvae with fluorescent macrophages, labelled with Tg(mpeg1:EGFP). Zebrafish were selected for imaging two hours after injection.	(0.4, 0.117, 0.117)	Light-Sheet microscopy	1	50 / - / -	1,108 / - / -

This dataset was curated in-house with the assistance of Cellpose 'cyto2' 2D model and u-Segment3D.

Supplementary Table 2. u-Segment3D settings and parameters for testing the reconstruction of 3D segmentation from ideal 2D segmentations with public datasets.

Supplementary Table 3. u-Segment3D settings and parameters for consensus 3D segmentation from the output of pretrained Cellpose 2D predictions with public datasets.

Supplementary Table 4. Cellpose2D and u-Segment3D settings and parameters for consensus 3D segmentation on additional demonstration datasets.

Supplementary Movies

Supplementary Movie 1. u-Segment3D enables consensus 3D segmentation from 2D segmented stacks

Supplementary Movie 2. Gradient descent dynamics of foreground cell coordinates using different 2D transforms.

Supplementary Movie 3. Comparison of the spatial proximity clustering used by Cellpose 3D mode and u-Segment3D's image-based connected component analysis on noisy cell tracking challenge datasets.

Supplementary Movie 4. Gradient descent dynamics of foreground cell coordinates of 3D single cells under 2D reconstructed 3D gradients.

Supplementary Movie 5. Gradient descent dynamics during 3D reconstruction of ovules, lateral root primordial and vasculature from ideal 2D segmented stacks.

Supplementary Movie 6. Segmentation of a movie of thin MDA231 human breast carcinoma cells embedded in collagen from the 3D cell tracking challenge using u-Segment3D to aggregate only 2D xy segmentations only.

Supplementary Movie 7. Segmentation of unwrapped surface cells of a Drosophila embryo over time using u-Segment3D to aggregate only 2D xy segmentations only.

Supplementary Movie 8. u-Segment3D postprocessing enables recovery of missing surface protrusions in the 3D segmentation of a HBEC cell aggregate.

Supplementary Movie 9. u-Segment3D postprocessing enables the segmentation of vessel sprouting in zebrafish.

Supplementary Movie 10. u-Segment3D segmentation of all cells in a $\approx 35\mu m$ x $1.5mm$ x $1.5mm$ CYCIF multiplexed tissue section of metastatic melanoma.

Supplementary Movie 11. u-Segment3D segmentation enabled detection of weakly fluorescent lung micrometastases in cleared tissues.

Supplementary Movie 12. u-Segment3D segmentation of heterogeneous cell structures in brain tissue labelled using coCATs.

References

1 Ma, J. *et al.* Segment anything in medical images. *Nature Communications* **15**, 654 (2024).

2 Ma, J. *et al.* The multi-modality cell segmentation challenge: Towards universal solutions. *arXiv preprint arXiv:2308.05864* (2023).

3 Schmidt, U., Weigert, M., Broaddus, C. & Myers, G. in *Medical Image Computing and Computer Assisted Intervention–MICCAI 2018: 21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part II 11.* 265-273 (Springer).

4 Graham, S. *et al.* CoNIC Challenge: Pushing the frontiers of nuclear detection, segmentation, classification and counting. *Medical image analysis* **92**, 103047 (2024).

5 Müller, A. *et al.* Modular segmentation, spatial analysis and visualization of volume electron microscopy datasets. *Nature Protocols*, 1-31 (2024).

6 Stringer, C., Wang, T., Michaelos, M. & Pachitariu, M. Cellpose: a generalist algorithm for cellular segmentation. *Nature methods* **18**, 100-106 (2021).

7 Cutler, K. J. *et al.* Omnipose: a high-precision morphology-independent solution for bacterial cell segmentation. *Nature methods* **19**, 1438-1448 (2022).

8 Wolny, A. *et al.* Accurate and versatile 3D segmentation of plant tissues at cellular resolution. *Elife* **9**, e57613 (2020).

9 Huang, Y. *et al.* Segment anything model for medical images? *Medical Image Analysis* **92**, 103061 (2024).

10 Tetteh, G. *et al.* Deepvesselnet: Vessel segmentation, centerline prediction, and bifurcation detection in 3-d angiographic volumes. *Frontiers in Neuroscience* **14**, 1285 (2020).

11 Segal, D. *et al.* In vivo 3D profiling of site-specific human cancer cell morphotypes in zebrafish. *Journal of Cell Biology* **221**, e202109100 (2022).

12 Lin, J.-R. *et al.* Multiplexed 3D atlas of state transitions and immune interaction in colorectal cancer. *Cell* **186**, 363-381. e319 (2023).

13 Haghighi, M., Caicedo, J. C., Cimini, B. A., Carpenter, A. E. & Singh, S. High-dimensional gene expression and morphology profiles of cells across 28,000 genetic and chemical perturbations. *Nature methods* **19**, 1550-1557 (2022).

14 Heemskerk, I. & Streichan, S. J. Tissue cartography: compressing bio-image data by dimensional reduction. *Nature methods* **12**, 1139-1142 (2015).

15 Zhou, F. Y. *et al.* Surface-guided computing to analyze subcellular morphology and membrane-associated signals in 3D. *bioRxiv*, 2023.2004. 2012.536640 (2023).

16 Carpenter, A. E. *et al.* CellProfiler: image analysis software for identifying and quantifying cell phenotypes. *Genome biology* **7**, 1-11 (2006).

17 Lalit, M., Tomancak, P. & Jug, F. Embedseg: Embedding-based instance segmentation for biomedical microscopy data. *Medical image analysis* **81**, 102523 (2022).

18 Weigert, M. *et al.* Content-aware image restoration: pushing the limits of fluorescence microscopy. *Nature methods* **15**, 1090-1097 (2018).

19 Archit, A. *et al.* Segment anything for microscopy. *bioRxiv*, 2023.2008. 2021.554208 (2023).

20 Israel, U. *et al.* A Foundation Model for Cell Segmentation. *bioRxiv* (2023).

21 Welf, E. S. *et al.* Quantitative Multiscale Cell Imaging in Controlled 3D Microenvironments. *Dev Cell* **36**, 462-475, doi:10.1016/j.devcel.2016.01.022 (2016).

22 Weems, A. D. *et al.* Blebs Promote Cell Survival by Assembling Oncogenic Signaling Hubs. *bioRxiv*, 2021.2004.2023.441200, doi:10.1101/2021.04.23.441200 (2021).

23 Drost, J. & Clevers, H. Organoids in cancer research. *Nature Reviews Cancer* **18**, 407-418 (2018).

24 Fatehullah, A., Tan, S. H. & Barker, N. Organoids as an in vitro model of human development and disease. *Nature cell biology* **18**, 246-254 (2016).

25 Yapp, C. *et al.* Multiplexed 3D Analysis of Cell Plasticity and Immune Niches in Melanoma. *bioRxiv* (2023).

26 Dutta, A. & Zisserman, A. in *Proceedings of the 27th ACM international conference on multimedia.* 2276-2279.

27 Torralba, A., Russell, B. C. & Yuen, J. Labelme: Online image annotation and applications. *Proceedings of the IEEE* **98**, 1467-1484 (2010).

28 Greenwald, N. F. *et al.* Whole-cell segmentation of tissue images with human-level performance using large-scale data annotation and deep learning. *Nature biotechnology* **40**, 555-565 (2022).

29 van Ineveld, R. L. *et al.* Revealing the spatio-phenotypic patterning of cells in healthy and tumor tissues with mLSR-3D and STAPL-3D. *Nature biotechnology* **39**, 1239-1245 (2021).

30 Maitin-Shepard, J. Neuroglancer. *github. com/google/neuroglancer, Retrieved*, 04-30 (2021).

31 Boergens, K. M. *et al.* webKnossos: efficient online 3D data annotation for connectomics. *nature methods* **14**, 691-694 (2017).
32 Hanslovsky, P. *et al.* (September, 2019).
33 Yushkevich, P. A., Gao, Y. & Gerig, G. in *2016 38th annual international conference of the IEEE engineering in medicine and biology society (EMBC).* 3342-3345 (IEEE).
34 Januszewski, M. *et al.* High-precision automated reconstruction of neurons with flood-filling networks. *Nature methods* **15**, 605-610 (2018).
35 Tasnadi, E. A. *et al.* 3D-Cell-Annotator: an open-source active surface tool for single-cell segmentation in 3D microscopy images. *Bioinformatics* **36**, 2948-2949 (2020).
36 Schmidt, M., Motta, A., Sievers, M. & Helmstaedter, M. RoboEM: automated 3D flight tracing for synaptic-resolution connectomics. *Nature Methods*, 1-6 (2024).
37 Shapson-Coe, A. *et al.* A connectomic study of a petascale fragment of human cerebral cortex. *BioRxiv*, 2021.2005. 2029.446289 (2021).
38 Dorkenwald, S. *et al.* Binary and analog variation of synapses between cortical pyramidal neurons. *Elife* **11**, e76120 (2022).
39 Consortium, M. *et al.* Functional connectomics spanning multiple areas of mouse visual cortex. *BioRxiv*, 2021.2007. 2028.454025 (2021).
40 Zheng, Z. *et al.* A complete electron microscopy volume of the brain of adult Drosophila melanogaster. *Cell* **174**, 730-743. e722 (2018).
41 Chen, J. *et al.* The Allen Cell and Structure Segmenter: a new open source toolkit for segmenting 3D intracellular structures in fluorescence microscopy images. *BioRxiv*, 491035 (2018).
42 Vicar, T. *et al.* Cell segmentation methods for label-free contrast microscopy: review and comprehensive comparison. *BMC bioinformatics* **20**, 1-25 (2019).
43 Wiesmann, V. *et al.* Using simulated fluorescence cell micrographs for the evaluation of cell image segmentation algorithms. *BMC bioinformatics* **18**, 1-12 (2017).
44 Wang, A. *et al.* A novel deep learning-based 3D cell segmentation framework for future image-based disease detection. *Scientific reports* **12**, 342 (2022).
45 Li, G. *et al.* 3D cell nuclei segmentation based on gradient flow tracking. *BMC cell biology* **8**, 1-10 (2007).
46 Marzec, M., Piórkowski, A. & Gertych, A. Efficient automatic 3D segmentation of cell nuclei for high-content screening. *BMC bioinformatics* **23**, 203 (2022).
47 Wu, T.-C., Wang, X., Li, L., Bu, Y. & Umulis, D. M. Automatic wavelet-based 3D nuclei segmentation and analysis for multicellular embryo quantification. *Scientific reports* **11**, 9847 (2021).
48 McDole, K. *et al.* In toto imaging and reconstruction of post-implantation mouse development at the single-cell level. *Cell* **175**, 859-876. e833 (2018).
49 Weigert, M., Schmidt, U., Haase, R., Sugawara, K. & Myers, G. in *Proceedings of the IEEE/CVF winter conference on applications of computer vision.* 3666-3673.
50 Wen, C. *et al.* 3DeeCellTracker, a deep learning-based pipeline for segmenting and tracking cells in 3D time lapse images. *Elife* **10**, e59187 (2021).
51 Dunn, K. W. *et al.* DeepSynth: Three-dimensional nuclear segmentation of biological images using neural networks trained with synthetic data. *Scientific reports* **9**, 18295 (2019).
52 Chen, H. & Murphy, R. F. 3DCellComposer-A Versatile Pipeline Utilizing 2D Cell Segmentation Methods for 3D Cell Segmentation. *bioRxiv*, 2024.2003. 2008.584082 (2024).
53 Kar, A. *et al.* Benchmarking of deep learning algorithms for 3D instance segmentation of confocal image datasets. *PLoS computational biology* **18**, e1009879 (2022).
54 Eschweiler, D., Rethwisch, M., Koppers, S. & Stegmaier, J. in *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI).* 792-796 (IEEE).
55 Smith, M. B. *et al.* Active mesh and neural network pipeline for cell aggregate segmentation. *Biophysical Journal* **122**, 1586-1599 (2023).
56 Cao, J. *et al.* Establishment of a morphological atlas of the Caenorhabditis elegans embryo using deep-learning-based 4D segmentation. *Nature communications* **11**, 6254 (2020).
57 Hardo, G., Noka, M. & Bakshi, S. Synthetic Micrographs of Bacteria (SyMBac) allows accurate segmentation of bacterial cells using deep neural networks. *BMC biology* **20**, 263 (2022).
58 Wu, L. *et al.* Nisnet3d: Three-dimensional nuclear synthesis and instance segmentation for fluorescence microscopy images. *Scientific Reports* **13**, 9533 (2023).
59 Martinez, N., Sapiro, G., Tannenbaum, A., Hollmann, T. J. & Nadeem, S. Impartial: Partial annotations for cell instance segmentation. *bioRxiv*, 2021.2001. 2020.427458 (2021).

60 Eschweiler, D., Rethwisch, M., Jarchow, M., Koppers, S. & Stegmaier, J. 3D fluorescence microscopy data synthesis for segmentation and benchmarking. *Plos one* **16**, e0260509 (2021).
61 Eschweiler, D. *et al.* Denoising diffusion probabilistic models for generation of realistic fully-annotated microscopy image datasets. *PLOS Computational Biology* **20**, e1011890 (2024).
62 Chai, B., Efstathiou, C., Yue, H. & Draviam, V. M. Opportunities and challenges for deep learning in cell dynamics research. *Trends in Cell Biology* (2023).
63 Kirillov, A. *et al.* in *Proceedings of the IEEE/CVF International Conference on Computer Vision.* 4015-4026.
64 Stringer, C. & Pachitariu, M. Cellpose3: one-click image restoration for improved cellular segmentation. *bioRxiv*, 2024.2002. 2010.579780 (2024).
65 Carreira, J. & Zisserman, A. in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 6299-6308.
66 Mazurowski, M. A. *et al.* Segment anything model for medical image analysis: an experimental study. *Medical Image Analysis* **89**, 102918 (2023).
67 Wang, Y., Zhao, Y. & Petzold, L. An empirical study on the robustness of the segment anything model (sam). *arXiv preprint arXiv:2305.06422* (2023).
68 Pachitariu, M. & Stringer, C. Cellpose 2.0: how to train your own model. *Nature methods* **19**, 1634-1641 (2022).
69 Berg, S. *et al.* Ilastik: interactive machine learning for (bio) image analysis. *Nature methods* **16**, 1226-1232 (2019).
70 Bui, N.-T., Hoang, D.-H., Tran, M.-T. & Le, N. Sam3d: Segment anything model in volumetric medical images. *arXiv preprint arXiv:2309.03493* (2023).
71 Jeong, W.-K. *et al.* Scalable and interactive segmentation and visualization of neural processes in EM datasets. *IEEE transactions on visualization and computer graphics* **15**, 1505-1514 (2009).
72 Funke, J. *et al.* Large scale image segmentation with structured loss based deep learning for connectome reconstruction. *IEEE transactions on pattern analysis and machine intelligence* **41**, 1669-1680 (2018).
73 Sonneck, J., Zhao, S. & Chen, J. in *Proceedings of the IEEE/CVF International Conference on Computer Vision.* 3894-3902.
74 Liu, Y., Jin, Y., Azizi, E. & Blumberg, A. J. Cellstitch: 3D cellular anisotropic image segmentation via optimal transport. *BMC bioinformatics* **24**, 480 (2023).
75 Driscoll, M. K. *et al.* Robust and automated detection of subcellular morphological motifs in 3D microscopy images. *Nat Methods* **16**, 1037-1044, doi:10.1038/s41592-019-0539-z (2019).
76 Michalska, J. M. *et al.* Imaging brain tissue architecture across millimeter to nanometer scales. *Nature Biotechnology*, 1-14 (2023).
77 Lam, L., Lee, S.-W. & Suen, C. Y. Thinning methodologies-a comprehensive survey. *IEEE Transactions on Pattern Analysis & Machine Intelligence* **14**, 869-885 (1992).
78 Tagliasacchi, A., Delame, T., Spagnuolo, M., Amenta, N. & Telea, A. in *Computer Graphics Forum.* 573-597 (Wiley Online Library).
79 Jain, A. K. *Fundamentals of digital image processing*. (Prentice-Hall, Inc., 1989).
80 Sethian, J. A. A fast marching level set method for monotonically advancing fronts. *proceedings of the National Academy of Sciences* **93**, 1591-1595 (1996).
81 Preibisch, S., Rohlfing, T., Hasak, M. P. & Tomancak, P. in *Medical Imaging 2008: Image Processing.* 128-135 (SPIE).
82 Preibisch, S. *et al.* Efficient Bayesian-based multiview deconvolution. *Nature methods* **11**, 645-648 (2014).
83 Daetwyler, S. *et al.* Imaging of cellular dynamics in vitro and in situ: from a whole organism to sub-cellular imaging with self-driving, multi-scale microscopy. *bioRxiv*, 2024.2002.2028.582579, doi:10.1101/2024.02.28.582579 (2024).
84 McInnes, L., Healy, J. & Melville, J. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. arXiv:1802.03426 (2018). <https://ui.adsabs.harvard.edu/abs/2018arXiv180203426M>.
85 Laine, R. F., Arganda-Carreras, I., Henriques, R. & Jacquemet, G. Avoiding a replication crisis in deep-learning-based bioimage analysis. *Nature methods* **18**, 1136-1144 (2021).
86 Nanes, B. A. *et al.* Keratin isoform shifts modulate motility signals during wound healing. *bioRxiv*, doi:10.1101/2023.05.04.538989 (2023).
87 Maška, M. *et al.* The Cell Tracking Challenge: 10 years of objective benchmarking. *Nature Methods* **20**, 1010-1020 (2023).
88 Cai, E. *et al.* Visualizing dynamic microvillar search and stabilization during ligand detection by T cells. *Science* **356**, eaal3118 (2017).

89 Petrie, R. J. & Yamada, K. M. At the leading edge of three-dimensional cell migration. *Journal of cell science* **125**, 5917-5926 (2012).
90 García-Arcos, J. M., Jha, A., Waterman, C. M. & Piel, M. Blebology: principles of bleb-based migration. *Trends in Cell Biology* (2024).
91 Weems, A. D. *et al.* Blebs promote cell survival by assembling oncogenic signalling hubs. *Nature* **615**, 517-525 (2023).
92 Mohan, A. S. *et al.* Enhanced Dendritic Actin Network Formation in Extended Lamellipodia Drives Proliferation in Growth-Challenged Rac1(P29S) Melanoma Cells. *Dev Cell* **49**, 444-460 e449, doi:10.1016/j.devcel.2019.04.007 (2019).
93 Rahaman, N. *et al.* in *International conference on machine learning.* 5301-5310 (PMLR).
94 Ke, L. *et al.* Segment anything in high quality. *Advances in Neural Information Processing Systems* **36** (2024).
95 Xu, C. & Prince, J. L. Snakes, shapes, and gradient vector flow. *IEEE Transactions on image processing* **7**, 359-369 (1998).
96 Lutton, J. E., Collier, S. & Bretschneider, T. A curvature-enhanced random walker segmentation method for detailed capture of 3D cell surface membranes. *IEEE Transactions on Medical Imaging* **40**, 514-526 (2020).
97 Zhou, D., Bousquet, O., Lal, T., Weston, J. & Schölkopf, B. Learning with local and global consistency. *Advances in neural information processing systems* **16** (2003).
98 He, K., Sun, J. & Tang, X. Guided image filtering. *IEEE transactions on pattern analysis and machine intelligence* **35**, 1397-1409 (2012).
99 Stringer, C. & Pachitariu, M. Transformers do not outperform Cellpose. *bioRxiv*, 2024.2004. 2006.587952 (2024).
100 Gaglia, G. *et al.* Lymphocyte networks are dynamic cellular communities in the immunoregulatory landscape of lung adenocarcinoma. *Cancer Cell* **41**, 871-886. e810 (2023).
101 Smits, J. P. H. *et al.* Immortalized N/TERT keratinocytes as an alternative cell source in 3D human epidermal models. *Sci Rep* **7**, 11838, doi:10.1038/s41598-017-12041-y (2017).
102 Rikken, G., Niehues, H. & van den Bogaard, E. H. Organotypic 3D Skin Models: Human Epidermal Equivalent Cultures from Primary Keratinocytes and Immortalized Keratinocyte Cell Lines. *Methods in molecular biology* **2154**, 45-61, doi:10.1007/978-1-0716-0648-3_5 (2020).
103 Ulman, V. *et al.* An objective comparison of cell-tracking algorithms. *Nature methods* **14**, 1141-1152 (2017).
104 Amat, F. *et al.* Fast, accurate reconstruction of cell lineages from large-scale fluorescence microscopy data. *Nature methods* **11**, 951-958 (2014).
105 Welf, E. S. *et al.* Quantitative multiscale cell imaging in controlled 3D microenvironments. *Developmental cell* **36**, 462-475 (2016).
106 Chang, B.-J. *et al.* Universal light-sheet generation with field synthesis. *Nature methods* **16**, 235-238 (2019).
107 Chang, B.-J., Dean, K. M. & Fiolka, R. Systematic and quantitative comparison of lattice and Gaussian light-sheets. *Optics express* **28**, 27052-27077 (2020).
108 Brand, M., Granato, M., Nüsslein-Volhard, C., Nusslein-Volhard, C. & Dahm, R. Zebrafish: A practical approach. *Keeping and raising zebrafish* **7**, 39 (2002).
109 Westerfield, M. The Zebrafish Book; A guide for the laboratory use of zebrafish (Danio rerio). *(No Title)* (2007).
110 Chi, N. C. *et al.* Foxn4 directly regulates tbx2b expression and atrioventricular canal formation. *Genes & development* **22**, 734-739 (2008).
111 White, R. M. *et al.* Transparent adult zebrafish as a tool for in vivo transplantation analysis. *Cell stem cell* **2**, 183-189 (2008).
112 Kaufmann, A., Mickoleit, M., Weber, M. & Huisken, J. Multilayer mounting enables long-term imaging of zebrafish development in a light sheet microscope. *Development* **139**, 3242-3247 (2012).
113 Daetwyler, S., Günther, U., Modes, C. D., Harrington, K. & Huisken, J. Multi-sample SPIM image acquisition, processing and analysis of vascular growth in zebrafish. *Development* **146**, dev173757 (2019).
114 Nirmal, A. J. *et al.* The spatial landscape of progression and immunoediting in primary melanoma at single-cell resolution. *Cancer Discovery* **12**, 1518-1541 (2022).
115 Lin, J.-R. *et al.* Highly multiplexed immunofluorescence imaging of human tissues and tumors using t-CyCIF and conventional optical microscopes. *elife* **7** (2018).
116 Quintana, E. *et al.* Human melanoma metastasis in NSG mice correlates with clinical outcome in patients. *Science translational medicine* **4**, 159ra149-159ra149 (2012).
117 Quintana, E. *et al.* Phenotypic heterogeneity among tumorigenic melanoma cells from patients that is reversible and not hierarchically organized. *Cancer cell* **18**, 510-523 (2010).

118 Dean, K. M. *et al.* Isotropic imaging across spatial scales with axially swept light-sheet microscopy. *Nature protocols* **17**, 2025-2053 (2022).
119 Marin, Z. *et al.* navigate: an open-source platform for smart light-sheet microscopy. *bioRxiv*, 2024.2002.2009.579083 (2024).
120 Yeo, I. K. & Johnson, R. A. A new family of power transformations to improve normality or symmetry. *Biometrika* **87**, 954-959 (2000).
121 Orieux, F., Giovannelli, J.-F. & Rodet, T. Bayesian estimation of regularization and point spread function parameters for Wiener–Hunt deconvolution. *JOSA A* **27**, 1593-1607 (2010).
122 Meijering, E. *et al.* Design and validation of a tool for neurite tracing and analysis in fluorescence microscopy images. *Cytometry Part A: the journal of the International Society for Analytical Cytology* **58**, 167-176 (2004).
123 Frangi, A. F., Niessen, W. J., Vincken, K. L. & Viergever, M. A. in *Medical Image Computing and Computer-Assisted Intervention—MICCAI'98: First International Conference Cambridge, MA, USA, October 11–13, 1998 Proceedings 1.* 130-137 (Springer).
124 Blum, H. A transformation for extracting new descriptions of shape. *Models for the perception of speech and visual form*, 362-380 (1967).
125 Tam, R. & Heidrich, W. in *IEEE Visualization, 2003. VIS 2003.* 481-488 (IEEE).
126 Zhang, T. Y. & Suen, C. Y. A fast parallel algorithm for thinning digital patterns. *Communications of the ACM* **27**, 236-239 (1984).
127 Goshtasby, A. A. Fusion of multi-exposure images. *Image and Vision Computing* **23**, 611-618 (2005).
128 Crouse, D. F. On implementing 2D rectangular assignment algorithms. *IEEE Transactions on Aerospace and Electronic Systems* **52**, 1679-1696 (2016).
129 Schindelin, J. *et al.* Fiji: an open-source platform for biological-image analysis. *Nature methods* **9**, 676-682 (2012).
130 Cignoni, P. *et al.* in *Eurographics Italian chapter conference.* 129-136 (Salerno, Italy).
131 Pettersen, E. F. *et al.* UCSF ChimeraX: Structure visualization for researchers, educators, and developers. *Protein science* **30**, 70-82 (2021).
132 Tofanelli, R., Vijayan, A., Scholz, S. & Schneitz, K. Protocol for rapid clearing and staining of fixed Arabidopsis ovules for improved imaging by confocal laser scanning microscopy. *Plant Methods* **15**, 1-13 (2019).
133 Willis, L. *et al.* Cell size and growth regulation in the Arabidopsis thaliana apical stem cell niche. *Proceedings of the National Academy of Sciences* **113**, E8238-E8246 (2016).
134 Refahi, Y. *et al.* A multiscale analysis of early flower development in Arabidopsis provides an integrated view of molecular regulation and growth control. *Developmental Cell* **56**, 540-556. e548 (2021).
135 Yang, J. *et al.* MedMNIST v2-A large-scale lightweight benchmark for 2D and 3D biomedical image classification. *Scientific Data* **10**, 41 (2023).
136 Yang, X., Xia, D., Kin, T. & Igarashi, T. in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 2656-2666.
137 Schneider, M., Reichold, J., Weber, B., Székely, G. & Hirsch, S. Tissue metabolism driven arterial tree generation. *Medical image analysis* **16**, 1397-1414 (2012).
138 Schneider, M., Hirsch, S., Weber, B., Székely, G. & Menze, B. H. in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2014: 17th International Conference, Boston, MA, USA, September 14-18, 2014, Proceedings, Part II 17.* 89-96 (Springer).