nature biotechnology

ARTICLES

https://doi.org/10.1038/s41587-021-01044-w

Check for updates

Cell segmentation in imaging-based spatial transcriptomics

Viktor Petukhov[1,2], Rosalind J. Xu[3,4,5], Ruslan A. Soldatov[1], Paolo Cadinu[3,4], Konstantin Khodosevich[2], Jeffrey R. Moffitt[3,4] and Peter V. Kharchenko[1,6] ✉

Single-molecule spatial transcriptomics protocols based on in situ sequencing or multiplexed RNA fluorescent hybridization can reveal detailed tissue organization. However, distinguishing the boundaries of individual cells in such data is challenging and can hamper downstream analysis. Current methods generally approximate cells positions using nuclei stains. We describe a segmentation method, Baysor, that optimizes two-dimensional (2D) or three-dimensional (3D) cell boundaries considering joint likelihood of transcriptional composition and cell morphology. While Baysor can take into account segmentation based on co-stains, it can also perform segmentation based on the detected transcripts alone. To evaluate performance, we extend multiplexed error-robust fluorescence in situ hybridization (MERFISH) to incorporate immunostaining of cell boundaries. Using this and other benchmarks, we show that Baysor segmentation can, in some cases, nearly double the number of cells compared to existing tools while reducing segmentation artifacts. We demonstrate that Baysor performs well on data acquired using five different protocols, making it a useful general tool for analysis of imaging-based spatial transcriptomics.

During the last decade, single-cell transcriptomic technologies gained great popularity, with single-cell RNA-sequencing (scRNA-seq) becoming the preferred approach for characterizing the state of complex tissues[1–4]. These techniques are being gradually augmented by spatially resolved transcriptomics measurements based on in situ sequencing[5,6], multiplexed single-molecule fluorescent in situ hybridization (smFISH)[7–9] or spatially barcoded hybridization[10,11]. The ability to examine the physical positions of different transcripts and cells at genomic scales has potential to bridge the molecular view of the cell with morphology, electrophysiology and other cellular phenotypes[12,13]. It can expose the impact of physical and biochemical interactions between cells and reveal how such processes influence tissue organization during development[14] and disease[15]. These protocols may eventually supplant scRNA-seq as they also offer technical advantages, such as the ability to bypass capricious tissue dissociation steps needed for scRNA-seq. At present, however, most such assays are limited in the number of genes they can detect (30–300 genes) as well as the number of molecules that can be detected per cell (50–500)[8,16]. Nevertheless, there has been steady progress on the optimization of these protocols, with some increasing the number of detectable genes to thousands[7,9,17,18]. Increasing scales and spatial resolution are already enabling unbiased characterization of tissue organization[19] and subcellular organization of cells[7,9,17].

The transcriptional data acquired by in situ sequencing or smFISH protocols can be generally summarized as a collection of detected molecules, each corresponding to a particular gene or transcript, along with the coordinates of that molecule within the field of view. While, in principle, such data can yield cellular or even subcellular resolution, the effective spatial resolution depends on the ability to distinguish features in the downstream analysis. Very sparse measurements, for instance, may only allow for interpretation of regional differences, such as segmentation of cortical layers. Achieving cellular resolution, however, even with high-density measurements, requires accurate cell segmentation. Most current groups have relied on an auxiliary nuclei staining (for example, DAPI) to identify putative cell centers[7,18]. But even such one-channel segmentation is challenging, commonly requiring manual tuning and corrections[20], including compensation for physical misalignment of molecular and auxiliary stains. Nuclei positions also do not inform on the extent of the cell body. Some efforts have used additional poly(A) staining to extend the initial nuclei positions[18,19]. Similarly, the probabilistic cell typing by in situ sequencing (pciSeq) algorithm[16] relies on the initial nuclei segmentation and extends the boundaries of the cell based on a Poisson model of gene expression. Alternatively, spatial measurements can be analyzed without explicit cell segmentation (segmentation free). Such approaches can characterize cell type composition of the tissue or identify distinct regions but cannot be easily extended to other kinds of analyses[14,21].

In this manuscript, we start by discussing the applications and limitations of the segmentation-free approach to imaging-based spatial transcriptomics. We suggest a new, simple method for segmentation-free analysis, which does not require extensive hyperparameter tuning. We then describe a general framework based on Markov random fields (MRFs) that can be used to solve a variety of molecule labeling problems, such as background separation or clustering. Following this framework, we introduce a method that performs cell segmentation based on the observed molecules and optional auxiliary staining data, modeling cells as smooth elongated distributions of distinct transcriptional composition. These methods are implemented in an open-sourced command line tool and a corresponding Julia package called Baysor. We show that Baysor can segment data from most published protocols with molecular resolution, yielding better segmentation accuracy and increasing the number of detected cells and the number of molecules in each cell.

[1]Department of Biomedical Informatics, Harvard Medical School, Boston, MA, USA. [2]Biotech Research and Innovation Centre, Faculty of Health and Medical Sciences, University of Copenhagen, Copenhagen, Denmark. [3]Program in Cellular and Molecular Medicine, Boston Children's Hospital, Boston, MA, USA. [4]Department of Microbiology, Harvard Medical School, Boston, MA, USA. [5]Department of Chemistry, Harvard University, Boston, MA, USA. [6]Harvard Stem Cell Institute, Cambridge, MA, USA. ✉e-mail: peter_kharchenko@hms.harvard.edu

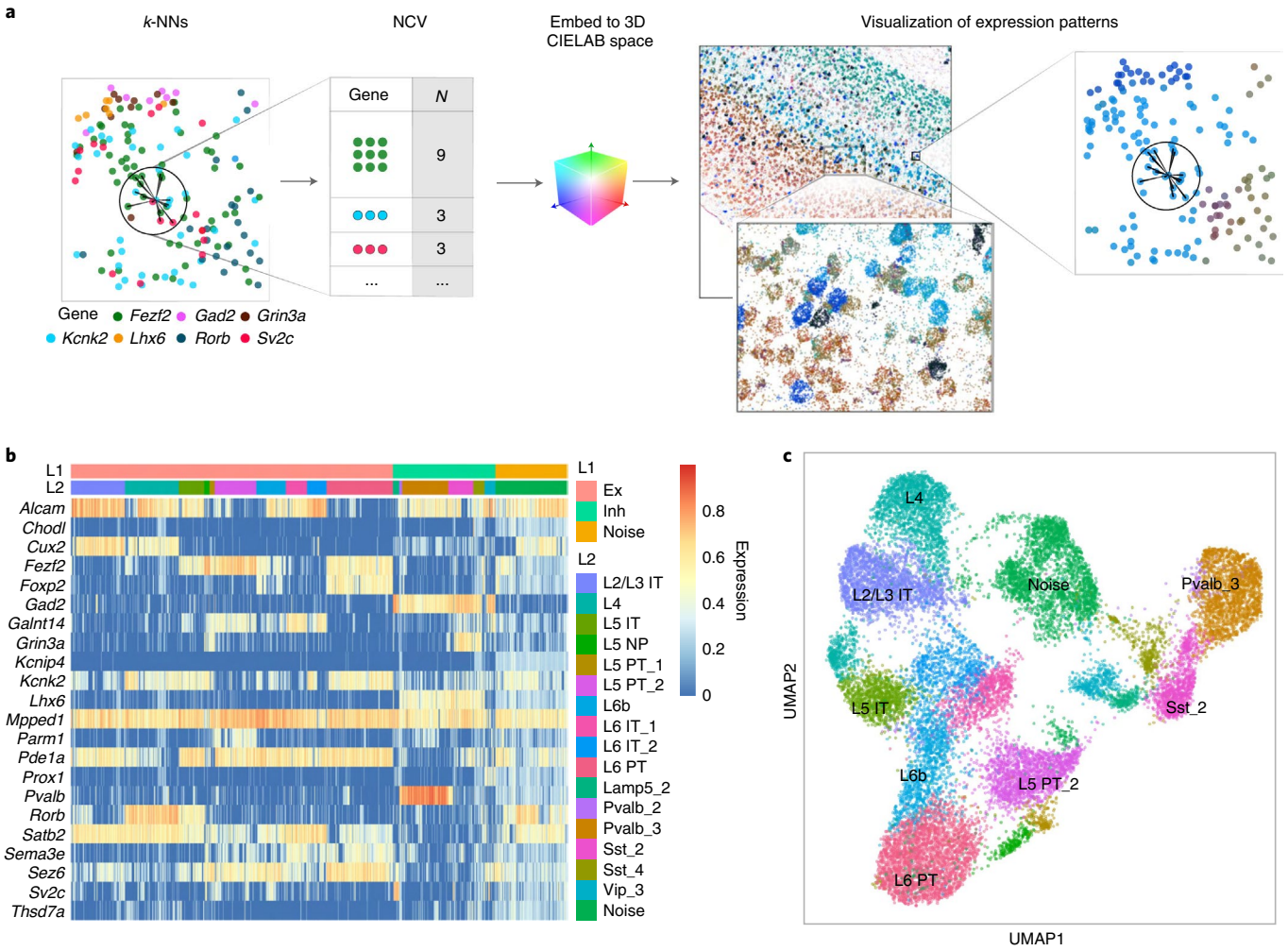


Fig. 1 | Segmentation-free analysis of spatial data using NCVs. a, NCVs are estimated by taking *k* spatially NNs for each molecule and tabulating the number of neighborhood molecules belonging to each gene (left). The molecules measured in the 2D space are shown as dots colored by the gene identity. Based on the similarity of these vector profiles, NCVs can be clustered or embedded into lower-dimensional space (**b**,**c**) (center, right). A 3D embedding can be translated into a color encoding so that similar colors correspond to similar neighborhood compositions. Such color encoding allows for effective visualization of individual cells as well as the overall tissue organization. A part of the Allen smFISH dataset is shown as an example. **b**, Heat map showing the expression patterns for 20,000 NCVs ($k=50$) that were uniformly sampled across the physical space with rows corresponding to genes and columns corresponding to NCVs. The color scale shows $\log_{10}$ of total count normalized expression additionally normalized by the maximum for each gene. The L1 and L2 column headers show the marker-based annotations for the corresponding NCVs; Ex, excitatory; Inh, inhibitory; IT, intratelencephalic; NP, near-projecting; PT, pyramidal tract. **c**, Uniform manifold approximation and projection (UMAP) embedding of NCVs colored by annotation.

Results

Analysis of local expression patterns without segmentation. As illustrated by the scRNA-seq studies, different cell types and many phenotypic states can be readily distinguished based on the transcriptional composition of a cell. In spatial measurements, the cells of a distinct type will give rise to small molecular neighborhoods with stereotypical transcriptional composition, making it possible to interpret such neighborhoods without performing explicit cell segmentation[21]. To capture this patch-like structure, we compute a neighborhood composition vector (NCV) for each molecule by taking its *k* spatially nearest neighbors (NNs) and estimating the relative frequency of different genes among the neighboring molecules (Fig. 1a). Embedding the total set of NCVs into a 3D color space results in a color encoding where the neighborhoods of similar transcriptional composition are represented by similar colors. When *k* is comparable to the size of the cell, under such color encoding, different types of cells and their boundaries become visually apparent (Fig. 1 and Supplementary Fig. 1).

The NCV expression vectors, in principle, can be treated as 'pseudo-cells' and analyzed using existing methods developed for scRNA-seq, including clustering, cell-type annotation and embedding (Fig. 1b,c and Supplementary Fig. 2a,b). However, because NCVs are generated for every molecule in the dataset, their absolute and relative abundance do not match the true composition of the tissue. NCVs of the molecules detected near cell boundaries may represent a mixture of distinct cell types, similar to doublets in scRNA-seq data (Supplementary Fig. 2c,d). For these reasons, below we develop more complex quantitative approaches for classifying and segmenting the measured molecules. The NCVs approach, however, provides effective and robust visualization of the transcriptional composition signal present in the data.

General approach for statistical labeling of spatial data. A number of analyses in spatial transcriptomics can be formulated as label assignment problems. Cell segmentation, for instance, assigns cell labels to the observed molecules. Similarly, separation of intercellular

background is a problem of labeling molecules as 'signal' versus 'background'. The distinguishing characteristics of these problems are that the labels tend to show strong spatial clustering; two nearby molecules, for instance, are likely to belong to the same cell and therefore share a common label. Mathematically, this spatial clustering tendency can be captured using MRF priors[22,23] on simple tessellation graphs (that is, without tuning of k). The labels themselves can be modeled as latent variables and inferred from the observed data using an expectation–maximization (EM) algorithm.

Different labeling problems can then be solved by choosing the appropriate label probability model and the observable data (Extended Data Fig. 1a). For instance, by using gene identities of the molecules as observables and multinomial distributions to model the transcriptional composition associated with different labels, this MRF-based approach results in a meaningful clustering of molecular neighborhoods (Fig. 2a,b). One can additionally use expression profiles of different cell types obtained from scRNA-seq data as priors for the multinomial distributions of different labels. This enables the approach to efficiently transfer the cell annotations from scRNA-seq to the measured molecules without performing cell segmentation (Supplementary Fig. 3). The MRF-based inference can be notably faster than traditional clustering (Supplementary Table 1). However, both performance and robustness of such an annotation transfer become poor when the number of cell types increases beyond 10–15, as the algorithm starts to capture stochastic fluctuations of the transcriptional composition. Another example of the labeling problem is distinguishing background molecules from cell bodies. In this setting, one can assume that the cells form dense regions, while the background noise molecules appear in sparse regions. Taking the distance to the k-th NN as a measure of sparsity (observed data), we used the same EM algorithm to segment the background (Fig. 2c,d and Extended Data Fig. 1a). Interestingly, across different datasets, we find that the transcriptional composition of the background molecules shows weak but noticeable regional variation, which is expected as extended cellular processes cannot be easily traced to the corresponding somas (Supplementary Fig. 2e–j). Overall, MRF provides a general recipe for solving a variety of spatial labeling problems, although each problem requires a custom formulation of the EM algorithm.

Cell segmentation across various protocols. Many of the downstream analyses and interpretations of the spatially resolved data depend on the ability to resolve individual cells. These include analysis of context-dependent cell expression states, physical interactions and spatial dependencies between cell types, cell migration and formation of tissue architecture. We, therefore, set out to develop a cell segmentation method that can take into account different facets of data that are informative of cell boundaries. The increased spatial density of molecules within the cell somas is one such facet, and the transcriptional composition of local molecular neighborhoods is another. Further evidence can be gained from stainings for nuclei (for example, DAPI), cell bodies (for example, poly(A)) or cellular membranes. To optimize cell segmentation based on multiple evidence sources, we have developed an algorithm, called Baysor, that builds on the ideas of the MRF segmentation outlined above. The method can be used to analyze data from various experimental protocols (Fig. 3) and can perform cell segmentation using molecular positions alone or by incorporating additional information. The approach models each cell as a distribution, combining spatial position and gene identity of each molecule. Thus, the whole dataset is considered as a mixture of such cell-specific distributions. Baysor then uses Bayesian mixture models (BMMs) to separate the mixture. The optimization relies on an MRF prior to ensure spatial separability of the cells and to encode additional information about the spatial relations of molecules (Extended Data Fig. 1b).

We used several strategies to evaluate the quality of computational segmentations, as it is currently challenging to establish ground truth. In tissues in which cells are not densely packed, staining of poly(A) can provide a way of estimating the extent of the cell[8,19]. As an initial benchmark, we evaluated the extent to which poly(A) boundaries are matched by the segmentations predicted by Baysor, the default DAPI Watershed method (using ImageJ[24]; see Methods) as well as a recently published pciSeq approach[16] (Extended Data Fig. 2). We found that Baysor predicted more cells and showed the lowest poly(A) signal intensity outside of the predicted cells. Overall, we find that both Baysor and pciSeq show good agreement with the poly(A)-predicted boundaries, while the Watershed approach performs notably worse (Extended Data Fig. 2d–f). Visualization of individual cells confirmed that while Watershed tends to underestimate the boundaries, Baysor segmentation is typically able to capture the extent of the cell soma (Extended Data Fig. 2g–i). In some examples, however, it was evident that the poly(A) signal itself can also be challenging to segment and can result in merges of cells with clearly distinct composition (Extended Data Fig. 2j,k).

While many existing cell segmentation methods rely on nuclear (DAPI) or cytoplasmic (poly(A)) staining[8,16,19], the challenges of obtaining accurate segmentation of such stains are well known[20,21], and the resulting calls typically contain errors. Such auxiliary stains, however, provide valuable information in cases where the molecular signal is sparse or not informative about cell boundaries (see Discussion). Baysor can take advantage of auxiliary stains by incorporating a precalculated segmentation as a probabilistic prior. To account for a limited degree of confidence in such segmentations, Baysor defines a 'prior segmentation confidence' parameter, which determines the weight of the prior. Setting this parameter to 0 will cause Baysor to ignore the prior, while a maximum value of 1 will restrict Baysor from changing segmentation of the molecules assigned to cells, leaving it to deal only with non-assigned molecules (Extended Data Fig. 3). Prior segmentation is also taken into account when determining the background to penalize removal of the molecules assigned to cells in the prior segmentation (see Methods). In addition to segmentation priors, Baysor can also incorporate information about background assignment probabilities per molecule. Finally, Baysor can use information about molecule clustering to penalize the assignment of molecules from different clusters to the same cell.

Next, we evaluated performance of Baysor and other segmentation methods on datasets generated using five different protocols[8,16,18,19] (Fig. 4). Examining summary statistics, we found that Baysor reported cells containing approximately the same number of molecules and area as the originally published ('Paper') segmentations, while Watershed and pciSeq[16] reported smaller segments, capturing mostly the molecules within the nucleus (Extended Data Fig. 4). Compared to the published segmentations, Baysor reported larger numbers of cells and higher fractions of molecules recognized as a part of a cell (Fig. 4a,b) with the largest, twofold difference for the osmFISH data[8]. Watershed and pciSeq underperformed by these two criteria as well. While the Baysor algorithm is stochastic in nature, the resulting segmentations are robust with respect to random starting points and parameter settings (Supplementary Fig. 4). Baysor normally relies on only one user-specified parameter (expected minimum number of molecules per cell) to determine all other settings (see Methods). The initial determination of molecular background probability is the most sensitive step and can be monitored visually using NCV visualization and diagnostic plots (Fig. 2d). We also profiled time and memory usage of the Baysor run, with the longest run taking 51 min for the MERFISH dataset with 3.7 million molecules and the largest memory usage of 40.4 GB for the STARmap dataset with 1,020 genes (Supplementary Table 2).

Given the challenges in establishing ground truth on cell segmentation, we devised a relative quality measure that examines

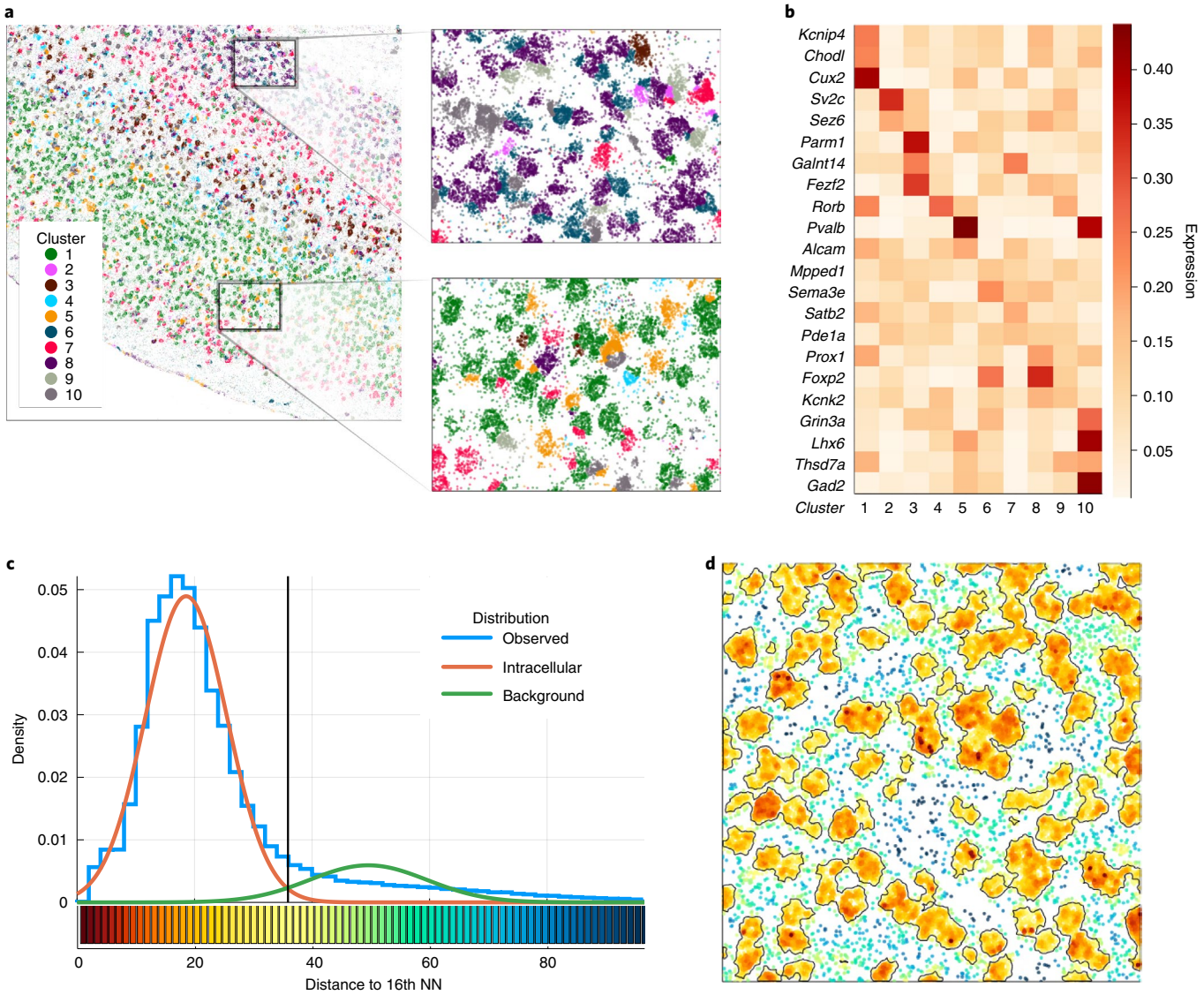


Fig. 2 | Application of an MRF framework for segmentation-free cell-type inference and background filtration. a,**b**, Individual molecules were clustered into major cell types by modeling the tissue as a mixture of multinomial distributions with an MRF prior. Cluster labels per molecule are shown in **a** with expression vectors for each of the clusters shown in **b**. **c**, The MRF approach is used to separate background from intracellular signal by modeling distance to its *k*-th nearest molecule (*x* axis, $k=16$) as a mixture of two normal distributions. Fitted intracellular and background distributions for the Allen smFISH dataset are shown in red and green, respectively. The vertical black line shows the optimal separation point. **d**, Molecules from a subset of the Allen smFISH dataset are shown as dots colored by their distance to the *k*-th NN, with the color key shown on the bottom of **c**. The black contours mark regions above 50% probability of being intracellular.

the differences between segmentations and evaluates which algorithm performs better in cases where the segmentations disagree. Specifically, in comparing any two segmentation results, we identified all cases where a cell from one segmentation ('Source') matched multiple cells from the other segmentation ('Target'). For each of these cases, we picked the largest part of the cell from the Source segmentation that matched to a single cell from the Target segmentation. We then estimated the correlation of expression profiles between this matching part and the rest of the Source cell (Fig. 4c). If the Source segmentation was correct, then the matching part should show similar transcriptional composition to the rest of the Source cell, and the resulting correlation measure will be high. By contrast, if the second (Target) segmentation was correct, the expression correlation will be low (Fig. 4d). For all protocols where the evaluation could be performed (see Methods), the overlapping regions showed, on average, higher expression correlation with the corresponding Baysor assignments than with the alternative segmentations, indicating higher accuracy of Baysor segmentation results than alternative segmentations, including Watershed, pciSeq and published segmentations (Fig. 4e–h and Supplementary Figs. 5–7).

We further investigated the two datasets where the differences between Baysor and published segmentations were most notable: the osmFISH[8] (Fig. 5) and MERFISH[19] (Extended Data Fig. 5) datasets. In both cases, the segmentation differences preferentially impacted certain cell types. In the case of osmFISH, the published segmentation omitted most of the cells of non-neuronal subtypes; only 10% of vascular and astrocytic cells detected by Baysor are present in the original segmentation (Fig. 5d). The disagreements on the MERFISH dataset were less biased, with the largest difference observed for endothelial cells and the published segmentation

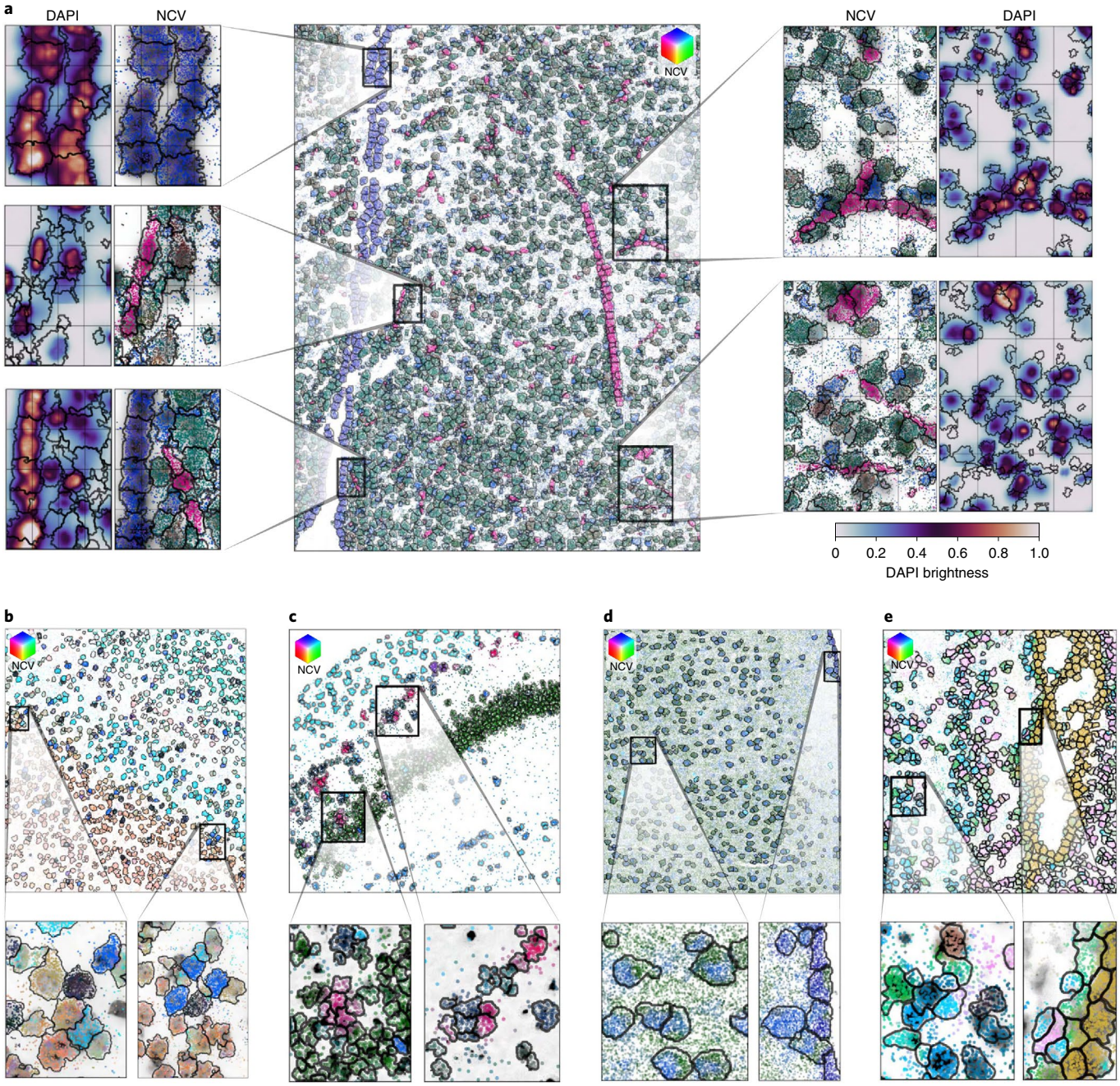


Fig. 3 | Examples of Baysor cell segmentation over the published protocols. a, Baysor segmentation is shown for a part of the MERFISH mouse hypothalamus[19] dataset. The center image shows positions of the measured molecules colored by their neighborhood gene composition (NCV; see Fig. 1a). The inferred boundaries of the segmented cells are shown as black contours. Zoom-in views are shown immediately to the left and right of the central plot (also colored by NCVs). The outer plots show DAPI signal within these regions. Additionally, the DAPI signal is shown as grayscale background within the zoom-in molecule plots. **b**–**e**, Additional examples of Baysor segmentation are shown for the Allen smFISH mouse VISp (**b**), ISS hippocampus[16] (**c**), STARmap mouse VISp 1020 (ref. [18]) (**d**) and ouroboros smFISH (osmFISH) somatosensory cortex[8] (**e**) datasets.

reporting 42% fewer cells (Extended Data Fig. 5c). This result was unsurprising given that the nucleus of many endothelial cells was not present within the physical slice as a consequence of the long cellular morphology of these cells. Without an imaged nucleus, such cells would be missed with the seeded Watershed method used in this work. There was also one subtype (ependymal cells) where Baysor distinguished 25% fewer cells. There, cells formed a homogeneous region with no signal to distinguish some of closely adjacent cells, which resulted in some level of undersegmentation (Extended Data Fig. 5g).

Membrane staining with MERFISH and 3D segmentation. While total poly(A) and DAPI staining can provide feature-rich costains suitable for segmentation in cell-sparse tissues such as the brain, such stains are not as useful for segmentation in cellular-dense tissues. To address this challenge, we have developed protocols to combine immunofluorescence (IF) of a pan-cell-type cell surface marker, the Na^+/K^+-ATPase, with MERFISH. Briefly, a secondary antibody to the anti-Na^+/K^+-ATPase primary antibody is labeled with a MERFISH readout sequence unique to that antibody[25]. Hybridization of the readout probe complementary

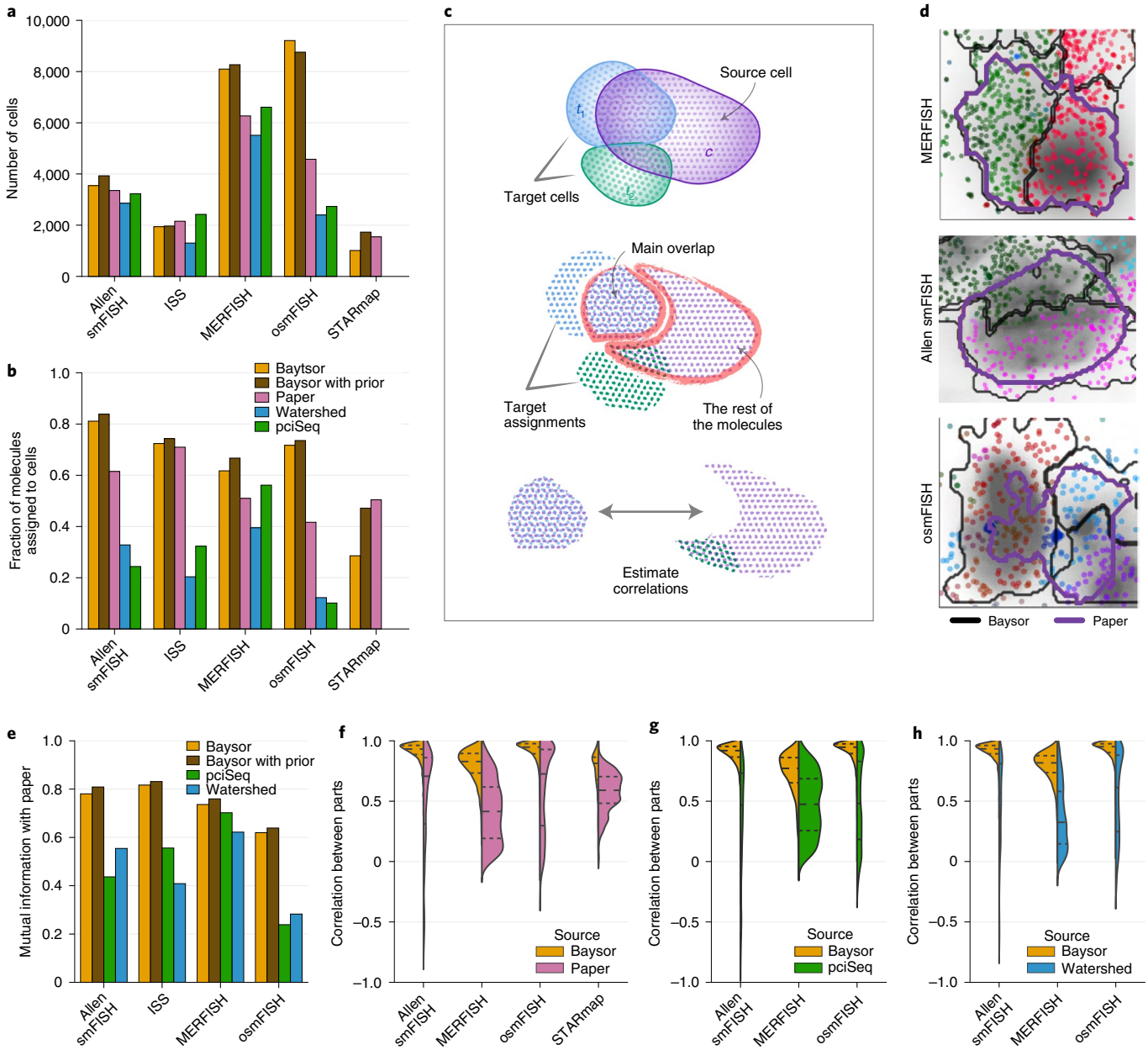


Fig. 4 | Comparison of Baysor segmentation with other methods and published results. a,**b**, Number of cells (**a**) and fraction of the molecules assigned to cells (**b**) by different segmentation methods (color) are shown for different datasets (*x* axis). **c**, Schematics for evaluating the differences between two segmentations based on gene composition of the results. Each cell from the Source segmentation *c* is matched to cells from the Target segmentation *t*. For the Source cells, which overlap multiple target cells, the region with the largest ('main') overlap is used. Correlation of gene expression of the main overlap region against the expression of the rest of the cell in the Source segmentation is then estimated. **d**, Examples of the results where the published segmentation merged distinct cell types. Dots show the measured molecules, colored by NCVs, with contours showing cell boundaries for Baysor (black) and reported in the original publications (purple). **e**, Agreement of different methods with the originally published segmentations measured as mutual information of molecule cell assignments (*y* axis). **f**, Comparison of Baysor results to the published segmentations using the correlation benchmark (**c**). The violin plots show the distribution of overlap correlations with the rest of the cell (*y* axis) for different datasets (*x* axis), with dashed lines representing quartiles of the distributions. The right side of each violin plot was calculated using Baysor segmentation as a Source and the published segmentation as a Target, while the left side of each plot was calculated by swapping the Source and Target segmentations. The width of the violin plots is proportional to the number of Source cells that were matched to multiple Target cells. The results show that in the regions of disagreement with the Paper segmentation, the overlap region correlates well with the rest of the Baysor cell, while the opposite is not always the case. **g**,**h**, Analogous to **f**, the plot compares performance of Baysor segmentation with the segmentation obtained using pciSeq (**g**) or Watershed (**h**) algorithms.

to this readout sequence reveals the locations of cell boundaries marked by Na^+/K^+-ATPase (Fig. 6a). By incorporating an acrydite moiety into the oligonucleotide that labels the antibody, the IF antibody signal can be embedded in the same polyacrylamide film that is used to stabilize mRNAs in the sample during tissue clearing[25,26].

We then used this cellular membrane IF–MERFISH protocol in the mouse small intestine to provide an additional benchmark

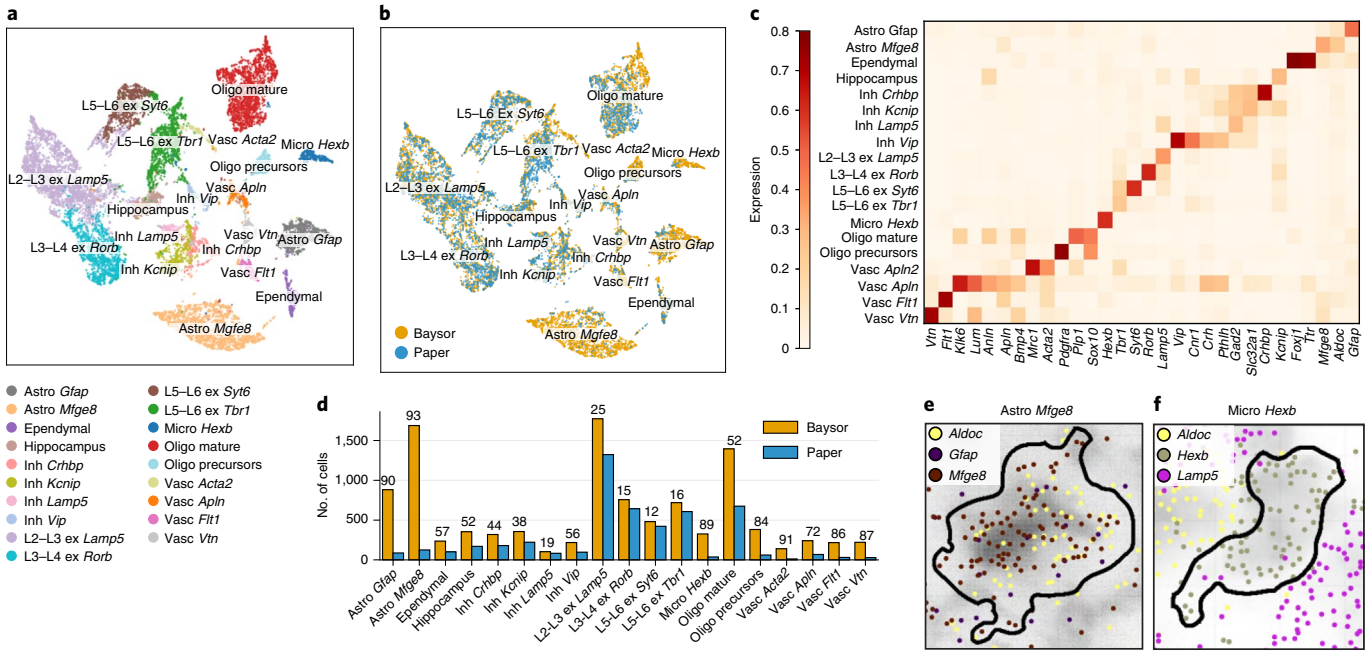


Fig. 5 | Examples of the segmentation differences on the osmFISH data. a, A joint UMAP embedding of the cells generated by both Baysor and the published segmentations labeling the annotated cell types with color; Astro, astrocyte; Inh, inhibitory; Ex, excitatory; Micro, microglia; Oligo, oligodendrocyte; Vasc, vascular. **b**, The same embedding colored by the segmentation method. **c**, A heat map showing expression patterns of marker genes (columns) for each of the cell types (rows). The colors show expression levels normalized by gene. **d**, Bar plots showing the number of cells per cell type for the Baysor (brown) and the published (blue) segmentations. Numbers on the top of the bars show excess percentage for the Baysor segmentation. **e**,**f**, Examples of Astro *Mfge8* (**e**) and Micro *Hexb* (**f**) cells, which were not segmented in the published segmentation but were distinguished using Baysor. The dots correspond to molecules colored by gene (only three of the most abundant genes are shown). The grayscale background represents DAPI signal, and the black line shows the cell boundary determined by Baysor.

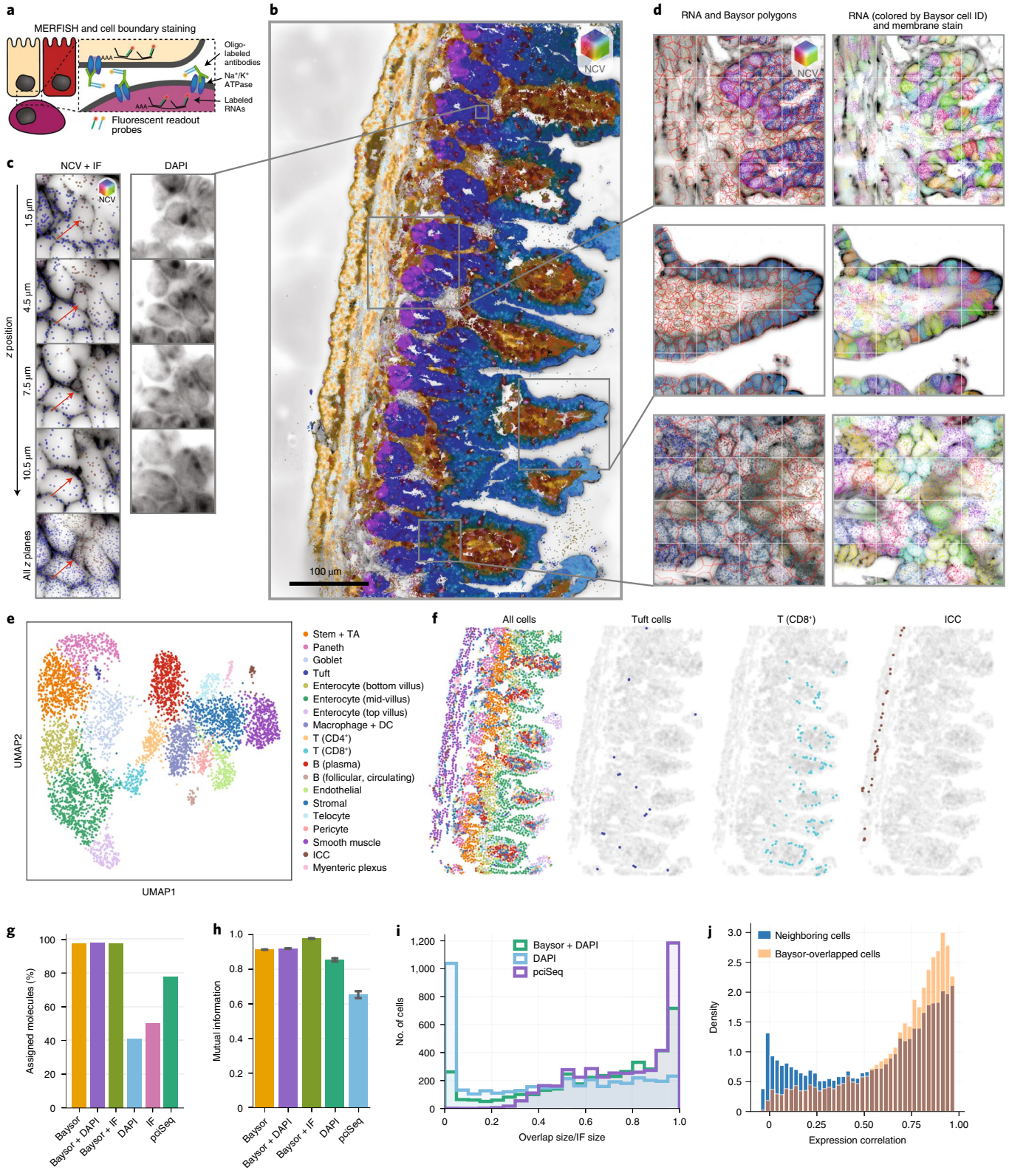
dataset with defined cell boundaries (Fig. 6b). We prepared cryosections of mouse ileum via a fresh–frozen protocol preceded by an mRNA preservation step in 4 mM ribonucleoside vanadyl complex (RVC). We then stained tissue sections with a MERFISH encoding probe library targeting 241 genes, including previously defined markers for the majority of gut cell types. Samples were also stained with anti-Na^+/K^+-ATPase primary antibodies, oligo-labeled secondary antibodies and DAPI. MERFISH measurements across multiple fields of view and nine *z* planes were performed to provide a volumetric reconstruction of the distribution of the targeted mRNAs, the cell boundaries marked by Na^+/K^+-ATPase IF and cell nuclei stained with DAPI (Fig. 6c).

This model dataset provided a useful setting for extending Baysor to dense and complex tissue types. First, in the crowded cellular environment of the gut, cell boundaries can vary greatly across the modest thickness of our slices (Fig. 6c). Second, the MERFISH library contained several genes with subcellular localization (for example, *Neat1*, which is nuclear localized, and *Apob*, *Slc5a1* and *Mptx2*, which are polarized within the cytoplasm). To address these challenges, we extended Baysor to perform segmentation in three dimensions and also added an option to incorporate prior knowledge of genes with pronounced intracellular mRNA localization (see Methods).

Application of Baysor to the 400 × 600 µm-sized ileum slice portion (Fig. 6b,d) partitioned mRNAs into cells that, when clustered, reproduced the rich diversity of cell types expected in the mouse ileum (Fig. 6e,f and Extended Data Figs. 6a and 7). In the epithelial layer, we identified intestinal stem cells and transit amplifying (TA) cells, various stages of enterocyte maturation along the villus, Paneth cells, goblet cells and the rare tuft cells; in the subepithelial compartments, we identified the expected diversity of immune cells, including B cells, T cells, macrophages and dendritic cells. In addition, we identified a range of stromal cells, including smooth muscle cells, ICCs, telocytes, pericytes, endothelial cells and cells associated with the enteric nervous system (Extended Data Figs. 6a and 7). Notably, Baysor was able to recover this cellular diversity in a dense and complex tissue without using the Na^+/K^+-ATPase membrane costain, further underscoring Baysor's ability to use the rich information contained within mRNA distributions.

As the Na^+/K^+-ATPase IF costain provides a high-accuracy independent assessment of the cell boundaries, we quantified the extent to which Baysor segmentation, as well as segmentations of other methods, agree with the IF membrane signal. Overall, we find that Baysor outperformed other methods (Fig. 6g–j and Extended Data Figs. 6 and 8). DAPI segmentations estimated by Cellpose[27] were within IF segments but, as expected, severely underestimated cell size. pciSeq appeared to overextend DAPI well beyond IF boundaries, reporting segments that were nearly twice as large as IF (Extended Data Fig. 8). Baysor also showed elevated cell size, although to a lesser extent. While Baysor showed best overall agreement with IF (Fig. 6h), there were some instances in which Baysor over- or undersegmented cell boundaries (Supplementary Fig. 8b and Extended Data Fig. 8). However, Baysor was able to recover more non-epithelial cells and revealed certain cell types, such as pericytes or telocytes, that were not apparent in the IF segmentation (Supplementary Fig. 8a and Extended Data Fig. 6c).

Outstanding challenges. Comparing between Na^+/K^+-ATPase stains, we noted that many instances where Baysor disagreed with the IF boundaries were from the regions in the gut that were defined by densely packed cells that are transcriptionally similar (Extended Data Fig. 9b). This situation is one in which RNA distributions



alone are unlikely to contain sufficient information to define cell boundaries. For the purposes of the downstream analyses, such uncertainty in the boundaries between cells of the same type is likely to be less consequential than other types of errors. Segmentation of such homotypic regions can be improved by incorporating priors on the boundaries defined from auxillary stains, such as Na^+/K^+-ATPase IF itself. When re-estimating cell boundaries with an IF segmentation prior, as expected, we find an excellent agreement between IF and Baysor segmentation (Extended Data Figs. 8 and 9). We find modest differences in the discovered cell types and their marker expression (Extended Data Fig. 6). A DAPI prior was not informative in this case due to dense cell packing (Extended Data Fig. 8). Using an IF prior does not reduce advantages of Baysor in other regions, as it still detects notably more cells than IF (Extended

Fig. 6 | Comparing Baysor segmentation to MERFISH measurements with stained cell boundaries in the mouse ileum. a, Cartoon representation of the process by which IF against a pan-cell-type cell surface marker, the Na^+/K^+-ATPase, is combined with MERFISH. **b**, Distribution of 241 RNAs measured with MERFISH in a slice of the mouse ileum. RNAs are colored by NCV; scale bar, 100 μm. Six replicate measurements from three different animals were performed with similar results. **c**, Zoom-in on a highlighted region in **b** showing the membrane stain (Na^+/K^+-ATPase IF) with RNAs (colored by NCV) and DAPI for select *z* planes. Bottom left, RNAs from all *z* planes are plotted on top of the IF image from the central *z* plane (6 μm). Arrows mark a cell visible only in a subset of *z* planes; scale bar, 20 μm. **d**, Representative zoom-in images from different regions of the ileum slice demonstrating the results of Baysor segmentation using RNA information alone. Left, polygon boundaries defined by Baysor (red lines), RNAs imaged across all nine *z* planes (colored by NCV) and the IF image from the central *z* plane (6 μm). Right, RNAs imaged across all nine *z* planes (colored such that RNAs assigned to the same cell have the same color) and the IF image from the central *z* plane. **e**, UMAP representation of the cells identified by Baysor colored by the results of Leiden clustering; TA, transit amplifying cells; DC, dendritic cells; ICC, interstitial cells of Cajal. **f**, Spatial distribution of all cell types (left) colored as in **e** and three representative rare cell types (right). **g**, Fraction of molecules annotated to cells in different segmentations. **h**, Agreement between different segmentations and membrane IF segmentation is assessed using mutual information across molecules for $n = 5$ central *z* planes. The average and 95% confidence intervals across *z* planes and dots for individual values are shown. Only molecules assigned in IF segmentations were used. **i**, For each cell of a given segmentation, the size of an overlap (in terms of molecules) of a best-matching membrane IF cell is shown as a fraction of IF cell size. **j**, Correlation of transcriptional composition is shown for all neighboring cells (blue) and for the cells that were joined in Baysor segmentation relative to membrane IF segmentation (orange). Baysor-joined cells tend to show high expression correlation.

Data Fig. 8 and Supplementary Fig. 8), including rare cell types that were not detected with IF segmentation alone (Extended Data Fig. 6).

The other challenge is posed by intracellular heterogeneity, which is particularly noticeable in highly multiplexed measurements possible with MERFISH[9,17] or sequential $FISH^+$ ($seqFISH^+$)[7]. This includes basic separation of the nuclear and cytoplasmic composition as well as more specialized polarization and compartmentalization patterns observed for subsets of transcripts in specific cell types (Extended Data Fig. 9a and 10). Baysor currently alleviates such issues by allowing for prior weights to be specified for polarized and nuclear-localized transcripts. Incidentally, nuclear localization priors also help to localize cell centers in homotypic and other challenging regions. In principle, it should be possible to automatically learn the stereotypical structure of intracellular heterogeneity using auxiliary stains, for instance a combination of DAPI and poly(A). The current Baysor model, however, assumes the cell body to be homogeneous. In the case of the STARmap dataset, inhomogeneity appeared to be technical in nature, with molecules belonging to the same gene exhibiting pronounced spatial clustering (Extended Data Figs. 10f and Supplementary Fig. 9). Baysor also currently assumes that cell shape can be reasonably well approximated using a multivariate normal prior, which can be a poor approximation for cells such as fibroblasts (Extended Data Fig. 10). Such issues can be currently mitigated by using more informative priors from auxiliary stainings, ensuring that the prior segmentation traces the complex cell shapes and is given higher prior weight.

Discussion

Realizing the potential of spatially resolved transcriptomics will require continued improvements on both the side of the protocols[12] and the side of the analytical methods for processing such data. Here, we focused on addressing an important preprocessing step of cell segmentation. Effective segmentation can increase the number of detected cells and provide more informative profiles for each cell. The accuracy of the segmentation is also critical for a number of valuable downstream inferences. For instance, incorrectly drawn borders can create spurious correlation of expression state between adjacent cell types, resulting in false-positive inference of cell interactions. Alternatively, shifted borders may be interpreted as transient cell states, suggesting false transitions between cell types. To avoid such potential issues, we described an approach that uses transcriptional composition to optimize the placement of cell boundaries in 2D or 3D. At its core, Baysor relies on a general MRF-based approach that can be used for solving other labeling problems on spatial data, such as separation of background molecules or clustering. Baysor can perform segmentation using only molecule placement data or in combination with evidence from auxiliary stains and yields improved segmentation quality, increased number of cells and segmented molecules.

Not all of the downstream analyses require cell segmentation. For instance, region segmentation or tissue cell-type composition may be inferred directly from molecular data[21]. We show that a relatively simple segmentation-free approach based on the composition of local neighborhoods (NCVs) can be used to assess the quality of the dataset, estimate the number and identity of the major cell types and effectively visualize the organization of the tissue (Fig. 1a). This approach is fast and does not require parameter tuning, making it a convenient option for preliminary analysis. Nevertheless, accurate cell segmentation opens up possibilities for quantitative modeling of tissue architecture and cell interactions (Fig. 6f).

Although the Baysor algorithm performed well on most of the published protocols, a number of potential improvements could be introduced, such as improving modeling of cell shapes[28]. Further improvements could be gained by extending the hierarchical Bayesian model to introduce cell-type-specific shape and composition characteristics, ideally incorporating explicit models of cell-type-specific transcript compartmentalization structure.

As we demonstrate, auxiliary stainings can be extremely valuable in resolving difficult cases. Improved stainings, such as labeling of outer membranes that we present here, as well as improved methods for segmenting such images will likely be key for improving the overall segmentation results. Both, however, face their own challenges. Manual processing is still often required to perform initial segmentation of DAPI and other common stains. Similarly, as we show, even membrane signals can be uneven across tissues and fail to confidently resolve boundaries of certain cells (Extended Data Fig. 6). It is therefore likely that optimal segmentations will rely on a combination of the transcriptional composition signal and information from auxiliary stains. As Baysor can take advantage of uncertain prior predictions, probabilistic auxiliary image segmentation methods would provide an advantage in that regard. We hope that Baysor implementation and the MRF-based computational approach will further facilitate the development and application of imaging-based spatial transcriptomics methods.

Online content

Any methods, additional references, Nature Research reporting summaries, source data, extended data, supplementary information, acknowledgements, peer review information; details of author contributions and competing interests; and statements of data and code availability are available at https://doi.org/10.1038/s41587-021-01044-w.

Received: 5 October 2020; Accepted: 4 August 2021;
Published online: 14 October 2021

References

1. Mereu, E. et al. Benchmarking single-cell RNA-sequencing protocols for cell atlas projects. *Nat. Biotechnol.* **38**, 747–755 (2020).
2. Regev, A. et al. The human cell atlas. *eLife* **6**, e27041. (2017).
3. HuBMAP Consortium. The human body at cellular resolution: the NIH Human Biomolecular Atlas Program. *Nature* **574**, 187–192 (2019).
4. Aldridge, S. & Teichmann, S. A. Single cell transcriptomics comes of age. *Nat. Commun.* **11**, 4307. (2020).
5. Lee, J. H. et al. Highly multiplexed subcellular RNA sequencing in situ. *Science* **343**, 1360–1363 (2014).
6. Ke, R. et al. In situ sequencing for RNA analysis in preserved tissue and cells. *Nat. Methods* **10**, 857–860 (2013).
7. Eng, C.-H. L. et al. Transcriptome-scale super-resolved imaging in tissues by RNA seqFISH. *Nature* **568**, 235–239 (2019).
8. Codeluppi, S. et al. Spatial organization of the somatosensory cortex revealed by osmFISH. *Nat. Methods* **15**, 932–935 (2018).
9. Xia, C., Fan, J., Emanuel, G., Hao, J. & Zhuang, X. Spatial transcriptome profiling by MERFISH reveals subcellular RNA compartmentalization and cell cycle-dependent gene expression. *Proc. Natl Acad. Sci. USA* **116**, 19490–19499 (2019).
10. Rodriques, S. G. et al. Slide-seq: a scalable technology for measuring genome-wide expression at high spatial resolution. *Science* **363**, 1463–1467 (2019).
11. Vickovic, S. et al. High-definition spatial transcriptomics for in situ tissue profiling. *Nat. Methods* **16**, 987–990 (2019).
12. Lein, E., Borm, L. E. & Linnarsson, S. The promise of spatial transcriptomics for neuroscience in the era of molecular cell typing. *Science* **358**, 64–69 (2017).
13. Bingham, G. C., Lee, F., Naba, A. & Barker, T. H. Spatial-omics: novel approaches to probe cell heterogeneity and extracellular matrix biology. *Matrix Biol.* **91-92**, 152–166 (2020).
14. Soldatov, R. et al. Spatiotemporal structure of cell fate decisions in murine neural crest. *Science* **364**, eaas9536 (2019).
15. Chen, W.-T. et al. Spatial transcriptomics and in situ sequencing to study Alzheimer's disease. *Cell* **182**, 976–991 (2020).
16. Qian, X. et al. Probabilistic cell typing enables fine mapping of closely related cell types in situ. *Nat. Methods* **17**, 101–106 (2020).
17. Chen, K. H., Boettiger, A. N., Moffitt, J. R., Wang, S. & Zhuang, X. RNA imaging. Spatially resolved, highly multiplexed RNA profiling in single cells. *Science* **348**, aaa6090 (2015).
18. Wang, X. et al. Three-dimensional intact-tissue sequencing of single-cell transcriptional states. *Science* **361**, eaat5691 (2018).
19. Moffitt, J. R. et al. Molecular, spatial, and functional single-cell profiling of the hypothalamic preoptic region. *Science* **362**, eaau5324 (2018).
20. Wang, Z. Cell segmentation for image cytometry: advances, insufficiencies, and challenges. *Cytometry A* **95**, 708–711 (2019).
21. Park, J. et al. Cell segmentation-free inference of cell types from in situ transcriptomics data. *Nat. Commun.* **12**, 3545 (2021).
22. Dirmeier, S. & Beerenwinkel, N. Structured hierarchical models for probabilistic inference from perturbation screening data. Preprint at *bioRxiv* https://doi.org/10.1101/848234 (2019).
23. Zhu, Q., Shah, S., Dries, R., Cai, L. & Yuan, G.-C. Identification of spatially associated subpopulations by combining scRNAseq and sequential fluorescence in situ hybridization data. *Nat. Biotechnol.* **36**, 1183–1190 (2018).
24. Rueden, C. T. et al. ImageJ2: ImageJ for the next generation of scientific image data. *BMC Bioinformatics* **18**, 529 (2017).
25. Wang, G., Moffitt, J. R. & Zhuang, X. Multiplexed imaging of high-density libraries of RNAs with MERFISH and expansion microscopy. *Sci. Rep.* **8**, 4847 (2018).
26. Moffitt, J. R. et al. High-performance multiplexed fluorescence in situ hybridization in culture and tissue with matrix imprinting and clearing. *Proc. Natl Acad. Sci. USA* **113**, 14456–14461 (2016).
27. Stringer, C., Wang, T., Michaelos, M. & Pachitariu, M. Cellpose: a generalist algorithm for cellular segmentation. *Nat. Methods* **18**, 100–106 (2021).
28. Yangel, B. & Vetrov, D. in *Energy Minimization Methods in Computer Vision and Pattern Recognition* (eds Heyden, A., Kahl, F., Olsson, C., Oskarsson, M., & Tai, X.-C.) p 137–150 (Springer, 2013).

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

© The Author(s), under exclusive licence to Springer Nature America, Inc. 2021

Methods

NCV analysis. To analyze the spatial expression patterns without cell segmentation, we use NCVs as a unit of analysis. NCVs are constructed by identifying k spatially NNs for each molecule and then characterizing its composition, that is, estimating the frequency of occurrences of different genes among the neighbors

$$NCV_i = \left\{ \frac{|u : (\text{gene}_u = q, u \in adj_k(i))|}{k} \right\}_{q=1:N_{\text{genes}}} \quad (1)$$

Here, N_{genes} is the total number of measured genes, gene_u is the gene that produced the molecule u, k is the number of NNs and $adj_k(i)$ are the indices of these NNs for the molecule i. To estimate the NNs, a k-d tree structure implemented in the NearestNeighbors.jl Julia package was used. Two-dimensional Euclidean distance was used as a distance metric. As implemented in the Baysor package, the default value of k was set to the expected minimal number of molecules per cell (a user-modifiable parameter) or the total number of detectable genes divided by 10, whatever is larger.

To perform scRNA-seq analysis of NCVs, we used the Pagoda2 (https://github.com/kharchenkolab/pagoda2) R package to calculate UMAP embedding[29] and the CellAnnotatoR package (https://github.com/khodosevichlab/CellAnnotatoR) to annotate the cell types.

To visualize the local gene composition, embedding of the NCVs into three dimensions was performed, first by reducing the dimensions using a principal-components analysis (PCA) to the top 15 principal components and then embedding the data into 3D space using UMAP with the default parameters min_dist = 0.1 and spread = 2.0. As fitting UMAP embedding for many NCVs is computationally intensive, to optimize performance, we first fit the UMAP embedding on a subset of NCVs and then applied the resulting transformation to all NCVs. In more detail, 10,000 NCVs were selected uniformly across the PCs by taking the sum over all PC coordinates for each of the NCVs, ranking them by the obtained values and then selecting a subset from this array uniformly across indices. Such an approach allows for a subset of NCVs with a density similar to the original distribution while avoiding stochastic sampling. After that, UMAP embedding was learned based on 10,000 selected NCVs in PC space. Next, the fitted UMAP embedding was used to project all PCA-transformed NCVs to 3D space. Finally, these 3D coordinates were renormalized and encoded into a perceptually uniform CIELAB color space. In a general case, normalization and encoding depends on the implementation for a specific library. Baysor delegates CIELAB encoding to the Colors.jl Julia package, which requires a^* and b^* components to be in the range of −100 to 100, while L^* component has to be in the range of 0 to 100. However, to avoid colors that are too close to black or white, Baysor restricts L^* component to a range of 10 to 90. If the assignment of molecules to background noise or true signal is available for a given dataset, only non-background molecules are used for fitting UMAP.

MRF for spatial segmentation. Here, we present a general probabilistic framework based on MRF applicable to a wide range of computational tasks in spatial transcriptomics. Observations about an RNA molecule, such as gene identity, spatial position or density of nearby RNA molecules, usually reflect latent objects, such as a cell or cell type of origin that produced the molecule or 'background'/'intracellular' compartment of the tissue. Thus, it is natural to consider a probabilistic model $P(\vec{x}, \vec{z})$ that models observed properties of molecules $\vec{x}$ and latent factors $\vec{z}$. Examples of pairs $(\vec{x}, \vec{z})$ could be observed density of nearby molecules and latent background/intracellular tissue regions for background separation problem, observed gene identities and latent cell types of origin for cell type segmentation problem or observed gene identity and position and latent cell of origin for cell segmentation problem. The probabilistic model $P(\vec{x}, \vec{z})$ is specified through parameters $\vec{v}$ of latent components with labels $\vec{z}$: $P(x_i, z_i = s|\vec{v}) = P(x_i|v_s) \cdot P(z_i = s)$. Following the traditional mixture model notation, here $P(z_i = s) = r_s$, such that $\sum_s r_s = 1$ are the mixture coefficients. Examples of $\vec{v}$ are parameters of expression levels for cell type segmentation or the expected molecule density for the corresponding regions for inferring the background/intracellular labels.

In our tasks, spatially proximal molecules usually share similar latent properties. For example, nearby RNA molecules likely come from the same cell. Traditional mixture models, however, do not necessarily account for this property, so we extended them with the MRF prior[22] $P_{\text{MRF}}(\vec{z})$ to formalize relationships of spatially close labels. It specifies dependence of a molecule label only on neighboring molecule labels: $P_{\text{MRF}}(z_i|\vec{z}_{-i}) = P_{\text{MRF}}(z_i|\vec{z}_k, k \in adj(i))$. Here, we rely on Potts MRF prior:

$$P_{\text{MRF}}(z_i|\vec{z}_k, k \in adj(i)) = \frac{1}{Z_i} \cdot \exp[\beta \sum_{j \in adj(i)} w_{ij}\delta(z_i, z_j)] \quad (2)$$

Here, w_{ij} is the MRF edge weight between molecules i and j (described below), β is a strength of the MRF prior, Z_i is a normalization constant and $\delta(z_i, z_j)$ is a delta function. Adding this prior, we get the probability for molecule i and component u:

$$P(\vec{x_i}, z_i = u|\vec{v}_u, r_u, \vec{z}_{-i}) = P(x_i|\vec{v}_u, z_i = u) \cdot P_{\text{MRF}}(z_i = u|\vec{z}_k, k \in adj(i)) \cdot r_u \quad (3)$$

Auxiliary information $\vec{a_v}$ about parameters $\vec{v}$, such as parameters of physical cell shape and cell type assignments, or prior segmentation labels are sometimes available in the form of additional priors $P(\vec{v}|\vec{a}_v)$ and $P(\vec{z}|\vec{a}_v)$ in the model. Additionally, a special case of prior information is using a Dirichlet prior over mixture probabilities $r_{z_i} \sim \text{Dir}(\alpha)$. Such a prior allows for inferring the number of mixture components automatically and is used in Cell segmentation. Thus, when prior information is available, the maximum likelihood estimates (MLEs) turn into maximum a posteriori (MAP) estimates for the following probability:

$$\begin{aligned} &P(\vec{x_i}, z_i = u|\vec{v}_u, \vec{a}_v, r_u, \vec{z}_{-i}) = P(x_i|\vec{v}_u, z_i = u) \cdot \\ &P(\vec{v}_u|\vec{a}_v) \cdot P_{\text{MRF}}(z_i = u|\vec{z}_k, k \in adj(i)) \cdot \\ &P(z_i = u|\vec{a}_v) \cdot r_{z_i} \cdot P(r_{z_i}|\vec{a}_v) \end{aligned} \quad (4)$$

Exact interpretation of parameters depends on the particular task (see Extended Data Fig. 1 for the graphical model representation); however, the notations stay the same. All vector variables are represented with lowercase letters accented by a right arrow (for example, $\vec{x}$), and matrix variables are shown in capital letters (for example, W). The following names are preserved across the further sections:

- $\vec{x}$ is the observed data used to infer the latent variables.
- $\vec{z}$ is the assignment of observations to components, with meaning of the components depending on the particular problem. It is a latent variable inferred by the algorithm.
- $\vec{v}_k$ is the vector of parameters for the component k. It is a latent variable inferred by the algorithm, and some prior information about these parameters is often available.
- $\vec{a}_v$ is the prior information about latent variables $\vec{v}_k$, $\vec{z}$ or r_k. This information may include uncertainty by itself and, in some cases, can be represented as a hierarchical Bayesian model.
- $W = \{w_{ij}\}$ is the matrix of edge weights of the MRF. It is considered as a meta-parameter of the algorithm. In practice, it is deterministically estimated from the data, although parameters of this estimation may vary.
- r_k is the mixing coefficient for the component k. It is inferred by the algorithm and, in most cases, it is proportional to the number of observations assigned to this component.

Variational EM inference. Parameter optimization in such a setting is commonly performed using the EM approach, iterating between E-step and M-step until convergence. E-step at iteration t estimates parameters of the posterior $P_t(\vec{z}) = P(\vec{z}|\vec{v}^t, \vec{x})$ following fixed parameter estimates $\vec{v}^t$, while M-step at iteration $t+1$ estimates $\vec{v}^{(t+1)}$ by maximizing expected likelihood of the observed data across the label posterior distribution P^t from the E-step $E_{\vec{z} \sim P^t} \log P(\vec{x}, \vec{z}|\vec{v})$.

In the case of MRF, however, posterior $P(\vec{z}|\vec{v}^t, \vec{x}) \sim P(\vec{x}|\vec{z}, \vec{v}) \cdot P_{\text{MRF}}(\vec{z})$ is intractable. We thus use variational EM[30], which approximates the posterior using mean field approximation $Q(\vec{z})$ that minimizes KL divergence $KL(Q(\vec{z})||P(\vec{z}|\vec{v}^t, \vec{x}))$ and factorizes over a set of latent factors

$$Q(\vec{z}) = \prod_i Q_i(z_i) \quad (5)$$

It can be shown[31] that factorized distribution $Q_i(z_i)$ has the following optimal form:

$$Q_i(z_i) \sim \exp(E_{-i}[\log P(z_i|\vec{z}_{-i}, \vec{x}, \vec{v})]) \quad (6)$$

Here, $\vec{z}_{-i}$ is a set of all latent factors except for z_i, and E_{-i} is expectation across $\vec{z}_{-i}$ drawn from $Q(\vec{z})$.

Because $P(z_i|\vec{z}_{-i}, \vec{x}, \vec{v}) \sim P(x_i|z_i, \vec{v}) \cdot P_{\text{MFR}}(z_i|\vec{z}_{-i}) \cdot r_{z_i}$, one can show that

$$\log Q_i(z_i) = \log P(x_i|z_i, \vec{v}) + \log r_{z_i} + E_{-i}\log P_{\text{MRF}}(z_i|\vec{z}_{-i}) + \text{const} \quad (7)$$

where the MFR term can be further decomposed as

$$\begin{aligned} &E_{-i}\log P_{\text{MRF}}(z_i|\vec{z}_{-i}) = E_{-i}\log\left(e^{\beta\sum_{j\in adj(i)} w_{ij}\delta(i,j)}\right) + \text{const} = \\ &\beta \cdot \sum_{j\in adj(i)} w_{ij}E_{-i}\delta(i,j) + \text{const} = \beta \cdot \sum_{j\in adj(i)} w_{ij}Q_j(z_i) + \text{const} \end{aligned} \quad (8)$$

E-step. The formulas above indicate that $Q_i(z_i)$ has the following update form in E-step when other latent variables are fixed:

$$Q_i^{(t+1)}(z_i) \sim P(x_i|z_i, \vec{v}^t) \cdot r_{z_i} \cdot e^{\beta \cdot \sum_{j\in adj(i)} w_{ij}Q_j^t(z_i)} \quad (9)$$

M-step. Without prior information, M-step of the variational EM estimates $\vec{v}_{t+1}$ by maximizing expectation of the likelihood $E_{\vec{z}\sim Q}\log P(\vec{x}, \vec{z}\,|\,\vec{v})$ across all labels $\vec{z}$ driven from Q:

$$\begin{aligned}
&E_{\vec{z}\sim Q}\log P(\vec{x}, \vec{z}\,|\,\vec{v}) = E_{\vec{z}\sim Q}\log P(\vec{x}\,|\,\vec{z}, \vec{v})\\
&+E_{\vec{z}\sim Q}\log P_{\mathrm{MRF}}(\vec{z}) + E_{\vec{z}\sim Q}\sum_i \log r_{z_i} =\\
&= \sum_{i=1}^{n_{\mathrm{mols}}} E_{\vec{z}\sim Q}\log P(x_i|z_i, v) + \sum_{i=1}^{n_{\mathrm{mols}}} E_{\vec{z}\sim Q} + \sum_{i,u} Q_i(u)\log r_u + \mathrm{const} =\\
&= \sum_{i=1}^{n_{\mathrm{mols}}}\sum_{u=1}^{n_{\mathrm{comps}}} (\log P(x_i|v_u) + \log r_u)\cdot Q_i(u) + \mathrm{const} =\\
&= \sum_{u=1}^{n_{\mathrm{comps}}}\left[\sum_{i=1}^{n_{\mathrm{mols}}} (\log P(x_i|v_u) + \log r_u)\cdot Q_i(u)\right] + \mathrm{const}
\end{aligned} \tag{10}$$

Thus, each variable v_u at M-step can be inferred by maximizing

$$\sum_{i=1}^{n_{\mathrm{mols}}} (\log P(x_i|v_u) + \log r_u)\cdot Q_i(u) \tag{11}$$

For the case of MAP estimates, the formula is adjusted with $P(v_u|\vec{a}_v)$ and $P(r_u|\vec{a}_v)$ (ref. [32])

$$\sum_{i=1}^{n_{\mathrm{mols}}} (\log P(x_i|v_u) + \log P(v_u|\vec{a}_v) + \log r_u + \log P(r_u|\vec{a}_v))\cdot Q_i(u) \tag{12}$$

Particular forms $P(x_i|v_u)$ and $P(v_u|\vec{a}_v)$ are specified for each application below. Label probabilities r_u are inferred identically for most applications unless explicitly specified:

$$r_u = \frac{\sum_i Q_i(u)}{\sum_{i,u'} Q_i(u')} \tag{13}$$

To summarize, variational EM consists of two steps:

- E-step: sequentially update distributions $Q_i^{(t+1)}(u)$ for each i upon fixed $\vec{v}^t$ and $Q_j^t(u)$, $j\neq i$ according to equation (9).
- M-step: update $\vec{v}$, $\vec{r}$ by maximizing equation (11) or equation (12).

Modeling background. The measurements we used for segmentation contain large amounts of background noise. Prior probability for each molecule to be recognized as signal ($p_{c,i}$) or background ($1-p_{c,i}$) is often available. The model above can be extended to incorporate such prior information without explicitly modeling background as a separate component. Here, we will show the formulas only for the case of the MLEs; however, it can naturally be extended for MAP estimates. Equation (3) implicitly assumes that all observations can be described by the specified model $P(x_i, z_i|\vec{v}_u, r_u, \vec{z}_{-i})$. Instead, we may assume that our data are generated by a two-level hierarchy of mixtures. The first level captures likely separation of background from signal where the probabilities are provided by the prior, and the second level is our mixture, where the probabilities need to be inferred.

In such a process, each molecule has a latent variable $\tau_i\in\{C, B\}$ showing whether this molecule was produced by one of the modeled components ($\tau_i=C$) or the background ($\tau_i=B$). Let us denote probability density of a molecule i produced by the background as $P_B(\vec{x_i}, z_i)$. In these definitions, equation (3) assumes that $\tau_i=C$ and so defines the probability $P(x_i, z_i|\vec{v}_u, r_u, \vec{z}_{-i}, \tau_i = C)$. Then, the probability density to observe a molecule i produced by the whole mixture is

$$\begin{aligned}
&P(x_i, z_i|\vec{v}, \vec{r}, \vec{z}_{-i}) =\\
&p(\tau_i = C)\cdot\sum_{u=1}^{n_{\mathrm{comps}}} P(x_i, z_i|\vec{v}_u, r_u, \vec{z}_{-i}, \tau_i = C) + p(\tau_i = B)\cdot P_B(\vec{x_i}, z_i)
\end{aligned} \tag{14}$$

Assuming the first-level coefficients are known ($p(\tau_i=C)=p_{c,i}$ and $p(\tau_i=B)=1-p_{c,i}$), we do not need to make any assumptions about the background density $P_B(\vec{x_i}, z_i)$. Instead, for the whole process described in Variational EM inference, we replace $P(x_i, z_i|\vec{v}_u, r_u, \vec{z}_{-i})$ with $P(x_i, z_i, \tau_i = C|\vec{v}_u, r_u, \vec{z}_{-i}) = P(x_i, z_i|\vec{v}_u, r_u, \vec{z}_{-i}, \tau_i = C)\cdot p(\tau_i = C)$. As a result, for each molecule i, its probability to belong to the component u is multiplied by $p(\tau_i=C)=p_{c,i}$: $\hat{Q}_i = Q_i\cdot p_{c,i}$. On the E-step equation (9), it affects the estimates of the MRF prior

$$\hat{Q}_i^{(t+1)}(z_i) \sim p_{c,i}\cdot P\left(x_i|z_i, \vec{v}^t\right)\cdot r_{z_i}\cdot e^{\beta\cdot\sum_{j\in adj(i)} w_{ij}p_{c,j}Q_j^t(z_i)} \tag{15}$$

On the M-step equation (11), it would correspond to maximizing the following for each v_u:

$$\sum_{i=1}^{n_{\mathrm{mols}}} (\log P(x_i, v_u) + \log r_u)\cdot p_{c,i}\cdot Q_i(u) \tag{16}$$

Introducing $p_{c,i}$ does not affect complexity of the algorithm from the computational point of view. On the E-step, we do not need to store $\hat{Q}_i$, as the algorithm works with non-normalized probabilities. During the MRF computations, $p_{c,j}$ can be incorporated into the MRF weights, transforming them as $\hat{w}_{ij} = w_{ij}\cdot p_{c,j}$. On the M-step, $p_{c,j}$ can be processed exactly in the same way as $Q_i(u)$, as it simply introduces additional weight to the observations. So, the equations for maximization stay the same with only the weights being updated.

Building the random field. To establish the structure of the random field in 2D space, Delaunay triangulation over points was built using the VoronoiDelaunay.jl package. It provides a connected planar graph matching the general structure of the space. For the 3D space triangulation, performance scales poorly, so a k-NN graph was used ($k=5$ by default). Then, in both cases, edge weights were set to the trimmed inverse Euclidean distance, so they represent connectivity of the two molecules i and j $w_{i,j} = min(q_{0.3}(\vec{d})/d_{i,j}, 1)$, where $d_{i,j} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$ and $q_{0.3}(\vec{d})$ is the 0.3 quantile of the distance distribution $\vec{d} = \{d_{i,j}\}$. In principle, the weights could be additionally adjusted to represent other kinds of dependencies between molecules. Details of such adjustments are described in the sections where they were used.

Applications. Below, we build probabilistic models and show variational EM updates for particular applications. One needs to specify the number of characteristics of the model in each of the applications.

- Observations x, number of labels z and parameters $\vec{v}_u$
- Generative model $P(x_i|\vec{v}_u)$
- Variation EM updates and initialization
- Priors $P(\vec{v}_u|\vec{a}_v)$ and/or $P(\vec{z}\,|\,\vec{a}_v)$ used
- Deviations from the model if there are any
- Stop criteria
- The way to build MRF if it is different from the description above

Separation of the intracellular molecules from the background. Observations x, reflecting local density of molecules around a given RNA molecule, are quantified as the distance to the k-th NN molecule (see Inferring the algorithm parameters for the k estimation). Label factors z correspond to 'background' and 'intracellular' regions of tissue. They are characterized by distributions of expected molecular densities $\mathcal{N}(\mu_c, \sigma_c)$ and $\mathcal{N}(\mu_b, \sigma_b)$ in background and intracellular regions, respectively.

The generative process. For observed distance x_i for a component $u\in\{c, b\}$,

$$p(x_i|z_i = u) \sim \mathcal{N}(\mu_u, \sigma_u) \tag{17}$$

E-step.

$$Q_i^{(t+1)}(u) \sim r_u\cdot\mathcal{N}(x_i|\mu_u, \sigma_u)\cdot e^{\beta\cdot\sum_{j\in adj(i)} w_{ij}Q_j^t(u)} \tag{18}$$

M-step.

$$\begin{aligned}
\mu_u &= \frac{\sum_{i=1}^{n_{\mathrm{mols}}} Q_i(u)\cdot x_i}{\sum_{i=1}^{n_{\mathrm{mols}}} Q_i(u)}\\
\sigma_u &= \frac{\sum_{i=1}^{n_{\mathrm{mols}}} Q_i(u)\cdot(x_i-\mu_u)^2}{\sum_{i=1}^{n_{\mathrm{mols}}} Q_i(u)}
\end{aligned} \tag{19}$$

Initialization. Initialization was performed using 10th and 90th percentiles of the distance distribution for the means of the intracellular (μ_c) and the background (μ_b) components correspondingly. Both standard deviations were initialized as $\sigma_c, \sigma_b = 0.25\cdot(\mu_b-\mu_c)$. The probability of a molecule to be labeled as intracellular (later called 'molecule confidence' for simplicity) was initialized as $Q_i^{(0)}(c) = \frac{\mathcal{N}(d_i|\mu_c,\sigma_c)}{\mathcal{N}(x_i|\mu_c,\sigma_c)+\mathcal{N}(x_i|\mu_b,\sigma_b)}$. The molecules from the right tail with $x_i>\mu_b+3\sigma$ were assigned to the background with probabilities fixed to 1.0 and excluded from subsequent optimization. Finally, the mixing coefficients are estimated according to equation (13). The β MRF parameter was set to 1 on all the performed experiments.

Using prior information. In case a prior segmentation $\overrightarrow{z^{\mathrm{prior}}}$ (for example, DAPI) was available, it was used as a constraint for the optimization. Given the prior segmentation confidence $c_{\mathrm{prior}}\in[0.0, 1.0]$, the expectation step was restricted for $Q_i(c) \geq c_{\mathrm{prior}}^2$ for all molecules assigned to cells in the prior segmentation ($z_i^{\mathrm{prior}} \neq b$)

$$\begin{aligned}
Q_i^{(t+1)}(c) &= c_{\mathrm{prior}}^2 + (1 - c_{\mathrm{prior}}^2)\\
&\cdot\frac{r_c\cdot\mathcal{N}(x_i|\mu_c,\sigma_c)\cdot e^{\beta\cdot\sum_{j\in adj(i)} w_{ij}Q_i^t(c)}}{\sum_{u\in\{c,b\}} r_u\cdot\mathcal{N}(x_i|\mu_u,\sigma_u)\cdot e^{\beta\cdot\sum_{j\in adj(i)} w_{ij}Q_i^t(u)}}\ \forall i:\ z_i^{\mathrm{prior}} \neq b\\
Q_i^{(t+1)}(b) &= 1 - Q_i^{(t+1)}(c)
\end{aligned} \tag{20}$$

At the limit of $c_{\text{prior}} \to 0$, this approach converges to the case without a prior segmentation, whereas $c_{\text{prior}} \to 1$ will ensure that the molecules assigned to cells in the prior segmentation will be necessarily recognized as intracellular. This part does not fit into the formal model described above, and it was chosen as it matches the general intuition of the prior segmentation confidence.

Stop criteria. The relative difference d of the parameters $\vec{\mu}$ and $\vec{\sigma}$ between iterations was used as the convergence criteria, with a convergence threshold of 0.005

$$d = \max\left(\frac{|\mu_c^{t+1} - \mu_c^t|}{\mu_c^t}, \frac{|\mu_b^{t+1} - \mu_b^t|}{\mu_b^t}, \frac{|\sigma_c^{t+1} - \sigma_c^t|}{\sigma_c^t}, \frac{|\sigma_b^{t+1} - \sigma_b^t|}{\sigma_b^t}\right) \quad (21)$$

After the algorithm converged, the MRF prior was discarded, and only the densities of the normal distribution were used to estimate the cell assignment probabilities

$$p_{c,i} = Q_i(c) = \frac{r_c \cdot \mathcal{N}(x_i|\mu_c, \sigma_c)}{r_c \cdot \mathcal{N}(x_i|\mu_c, \sigma_c) + r_b \cdot \mathcal{N}(x_i|\mu_b, \sigma_b)} \quad (22)$$

This was done because the MRF prior consistently pushes probabilities to be close to either 0.0 or 1.0, which corresponds to binary classification. By contrast, using only normal densities results in more gradual probability values, which can be integrated better with the further probabilistic parts of the algorithm.

Segmentation of cell types (molecule clustering). Each observation x_i is the gene identity of a molecule. Label factors z correspond to cell types, and the number of cell types n_{comps} is a fixed meta-parameter. Cell-type labels are characterized by categorical distributions $\text{Cat}(x_i|\vec{v}_u)$ for each cell type u with a vector of gene probabilities $\vec{v}_u$.

The generative process. The generative process for observed gene x_i is

$$p(x_i|z_i = u) \sim \text{Cat}(x_i|\vec{v}_u) \quad (23)$$

E-step.

$$Q_i^{(t+1)}(z_i) \sim \text{Cat}(x_i|\vec{v}_{z_i}) \cdot e^{\beta \cdot \sum_{j \in adj(i)} w_{ij} \cdot p_{c,j} \cdot Q_j^t(z_i)} \quad (24)$$

Here, $p_{c,j}$ represents the molecule confidences estimated on the previous step incorporated according to Modeling background. The mixing coefficients r_{z_i} were deliberately removed, which corresponds to the assumption that all mixture components have the same coefficients $r_u = \frac{1}{n_{\text{comps}}}\ \forall u$, regardless of the number of observations per component. This was done because having such a dependency forced the algorithm to prefer components of larger size, and, without strong signal from data, the MRF prior tended to eliminate all but the largest component. Such problems do not arise in the previous application because for molecular densities, the two normal distributions described the observed data much better than one, ensuring that both components (background and signal) were maintained. By contrast, here, the whole dataset can be described as a single categorical distribution with a high quality of fit, pushing the algorithm to converge on one component.

M-step. M-step is as follows with the justification described below:

$$v_{u,q} = \frac{\sum_{i:g_i=q}[Q_i(u) \cdot p_{c,i}] + 1}{\sum_{i=1}^{n_{mols}}[Q_i(u) \cdot p_{c,i}] + 1} \quad (25)$$

The conventional rule for updating the parameters of a categorical distribution is $v_{u,q} = \frac{\sum_{i:g_i=q} Q_i(u)}{\sum_{i=1}^{n_{mols}} Q_i(u)}$. However, using it can be problematic because if a gene g was not previously observed for a specific component u, then its probability always remains zero ($v_{u,g}=0$). A popular solution for this is Laplace smoothing (also known as pseudocounts), $v_{u,q} = \frac{\sum_{i:g_i=q} Q_i(u) + \alpha}{\sum_{i=1}^{n_{mols}} Q_i(u) + \alpha \cdot n_{\text{genes}}}$, which also corresponds to using a symmetric Dirichlet prior with parameter α. But, Laplace smoothing introduces dependency on the total number of genes n_{genes} in the dataset, which is undesired for categorical mixtures with sparse probability vectors. This problem becomes apparent when considering a dataset with d possible gene types, some gene $g \in 1{:}d$. Let us consider two categorical components $C_1(g)$ and $C_2(g)$, which have v_1 and v_2 molecules of g assigned to them with N_1 and N_2 total molecules assigned correspondingly. If we estimate the likelihood ratio for these two components with Laplace smoothing, it is equal to $\frac{(v_1+1)\cdot(N_2+d)}{(v_2+1)\cdot(N_1+d)}$. In realistic scenarios, $d > N_u \gg v_u$, so the ratio is dominated by d and is always close to 1. To avoid this, we replaced d in the Laplace smoothing with a constant 1 $v_{u,q} = \frac{\sum_{i:g_i=q} Q_i(u) + 1}{\sum_{i=1}^{n_{mols}} Q_i(u) + 1}$. This results in non-normalized probability densities, but this fact does not affect the rest of the algorithm. Then, these formulas are further adjusted by molecule confidences according to Modeling background, which leads to equation (25).

Using prior information. This model can be further enhanced by incorporating prior information about transcriptional composition of cell clusters (for example, scRNA-seq cell types). The method we used is ad hoc without a solid mathematical foundation, which deviates from the general model described above. If prior information is available, the maximization step is then changed to take it into account. Given prior expression fraction $\mu_{u,q}$ from the cluster u and the gene q as well as its standard deviation $\sigma_{u,q}$ (both can be estimated from scRNA-seq data), the estimate was adjusted based on the z score value $\zeta_{u,q} = \frac{v_{u,q} - \mu_{u,q}}{\sigma_{u,q}}$ as

$$v'_{u,q} = \begin{cases} v_{u,q}, & \text{if } |\zeta_{u,q}| < 1 \\ \mu_{u,q} + \text{sign}(\zeta_{u,q}) \cdot \left(\frac{|\zeta_{u,q}|}{4} + 0.75\right) \cdot \sigma_{u,q}, & \text{if } 1 \le |\zeta_{u,q}| < 3 \\ \mu_{u,q} + \text{sign}(\zeta_{u,q}) \cdot \left(\sqrt{|\zeta_{u,q}|} + 1.5 - \sqrt{3}\right) \cdot \sigma_{u,q}, & \text{if } |\zeta_{u,q}| \ge 3 \end{cases} \quad (26)$$

This function applies no penalty for all deviation from the mean within one standard deviation, linear penalty for deviations less than three standard deviations and a super-linear penalty otherwise (Supplementary Fig. 10a). If the standard deviation was not available, it was set to the mean value by default, $\sigma_{u,q} = \mu_{u,q}$. It is important to note here that for a given prior clustering, reasonable results can often be obtained by running only the expectation step without the maximization, which corresponds to setting $\sigma_{u,q} = 0, \forall\, u, q$. Particularly, the results shown in Supplementary Fig. 3 were obtained iterating only over the expectation step.

Initialization. To provide a reasonable starting point for optimization, the algorithm takes into account local gene coexpression structure. To do so, we first estimate the coexpression matrix $S = \{s_{i,j}\}_{i,j \in 1:n_{\text{genes}}}$, estimating coexpression level between genes q_1 and q_2 as a fraction of weights between all molecules of q_1 and q_2, connected with an MRF edge, relative to the geometric average between total weights of edges from molecules of q_1 and molecules of q_2

$$s_{q_1,q_2} = \frac{\sum_{i:g_i=q_1}\sum_{j \in adj(i), g_j=q_2} w_{i,j}}{\sqrt{\left(\sum_{i:g_i=q_1}\sum_{j \in adj(i)} w_{i,j}\right) \cdot \left(\sum_{i:g_i=q_2}\sum_{j \in adj(i)} w_{i,j}\right)}} \quad (27)$$

Then, the independent component analysis (ICA) is applied to the matrix S (implementation from MultivariateStats.jl) with the number of components equal to the desired number of clusters. The absolute values of the loading matrix W are used as initial expression values $v_{u,q}^{\text{init}} = |W_{u,q}|$. In case the ICA does not converge, the algorithm is initialized using a vector of gene frequencies estimated over the whole dataset multiplied by uniform noise in [0.95; 1.05]: $v_{u,q}^{\text{init}} = \frac{|\{i:\, g_i = q\}| \cdot \text{Unif}(0.95, 1.05)}{n_{\text{mols}}}$. Finally, these values were normalized by the sum over all genes $v_{u,q} = \frac{v_{u,q}^{\text{init}}}{\sum_{i=1}^{n_{\text{genes}}} v_{u,i}^{\text{init}}}$. The initial assignment probabilities were estimated as $p_{u,i} = \frac{\text{Cat}(g_i|\vec{v}_u)}{\sum_{s=1}^{n_{\text{comps}}} \text{Cat}(g_i|\vec{v}_s))}$. The MRF β parameter was set to 1 on all the performed experiments.

Stop criteria. Convergence was determined based on the maximal change in $p_{u,i}$ between iterations weighted by $p_{c,i}$ $\max_{\substack{1 \le i \le n_{\text{mols}} \\ 1 \le u \le n_{\text{comps}}}} \left(|p_{u,i}^{(t)} - p_{u,i}^{(t-1)}| \cdot p_{c,i}\right)$. The default threshold for the convergence was set to 0.01. The algorithm stops when the condition is satisfied for 20 continuous iterations.

Compartment segmentation. Sometimes, gene expression data can have intracellular structure with few genes specific only for some cell compartments. The model here is described in terms of separation of nuclei and cytoplasm; however, it can be easily extended to describe other kinds of compartments, such as mitochondria. It is important to note, however, that compartment-specific genes are often also cell-type specific. So besides labeling the compartment wherever it is possible, the algorithm must also recognize regions where no compartmentalization is observed. The input of the algorithm in this case is a set of genes that are specific to nuclei or cytoplasm (at least one gene per compartment). The goal of the algorithm is to infer compartment localization for molecules for other genes for which compartment information has not been provided. Given the graphical structure of the data, this problem is known in general case as information propagation on a graph. The approach we picked relies on the same Potts model and can be considered as a degenerate case of the general model described above (MRF for spatial segmentation). Our observed data in this case are labeling for molecules $x_i \in$ {'nuclei', 'cyto', 'na', 'unknown'} (see 'Initialization' step). The components are $u \in$ {'nuclei', 'cyto', 'na'} with no parameters to fit.

The generative process. The following is the generative process for observed label x_i:

$$p(x_i|z_i = u) = \begin{cases} 0.5, & \text{if } x_i = u \text{ or } x_i = \text{unknown} \\ 0, & \text{otherwise} \end{cases} \quad (28)$$

E-step.

$$Q_i^{(t+1)}(z_i) \sim \begin{cases} e^{\beta \cdot \sum_{j \in adj(i)} w_{ij} \cdot p_{c,j} \cdot Q_j^t(z_i)}, & \text{if } x_i = \text{unknown} \\ Q_i^t(z_i), & \text{otherwise} \end{cases} \quad (29)$$

M-step. M-step is degenerate because there are no parameters to optimize.

Initialization. Given lists of genes per compartment l_{nuclei} and l_{cyto}, the algorithm assigns all molecules produced by genes from a list l_u to the label u. Next, for each molecule i, the distances to the second nearest molecule from each compartment (d_i^{nuclei} and d_i^{cyto}) are estimated. The maximum across these distances $d_i = \max(\overset{nuclei}{d}_i, d_i^{cyto})$ is used to determine whether a molecule belongs to a non-compartmentalized region. All molecules with $d_i \geq 3s$ are assigned to the 'na' class. The assignment probabilities $Q_i^{(0)}(z_i)$ for the labeled molecules are fixed to 1 for the corresponding class. The other molecules are assigned to the 'unknown' class, and their assignment probabilities are initialized as

$$Q_i^{(0)}(u) = \begin{cases} \max\left(\min\left(0.5 \cdot \left(\frac{d_i}{s} - 1\right), 1\right), 0\right), & \text{if } u = \text{na}, \\ \frac{1 - Q_i^{(0)}(u=\text{na})}{2}, & \text{otherwise} \end{cases} \quad (30)$$

For the presented results, the MRF β parameter was set to 1.

Stop criteria. The stop criteria are exactly the same as for the cell type segmentation. Convergence was determined based on $\max_{\substack{1 \leq i \leq n_{\text{mols}} \\ 1 \leq u \leq n_{\text{comps}}}} \left(|p_{u,i}^{(t)} - p_{u,i}^{(t-1)}| \cdot p_{c,i}\right)$. The default threshold for the convergence was set to 0.01. The algorithm stops when the condition is satisfied for 20 continuous iterations.

Cell segmentation. For each RNA molecule i, we record gene identity g_i, spatial positions (in 2D or 3D) $\vec{x}_i$ as well as the probability that the molecule does not belong to the background $p_{c,i}$ ('molecule confidence'; see equation (22)). When available, the information about cell type per molecule cl_i is also used. An optional prior cell segmentation $\vec{z}^{\text{prior}}$, optional information about compartment-specific expression of select genes as well as a probability of a molecule being found in nuclei $p_{n,i}$ can also be taken into account in an ad hoc manner (see Using a prior segmentation). The goal of this stage is to infer molecule cell assignment z_i. It is important to note that while the algorithm here uses results from the segmentation problems described above, most of the assigned labels (such as cell types or compartments) are treated as fixed observed information but not as latent variables (as was done previously). The only exceptions are molecule confidence values, which use raw probabilities but not hard assignment. So, no other latent labels on the molecule level are estimated.

Each cell u is described by its cell shape and modeled as an ellipsoid using 2D or 3D Gaussian distribution with parameters $\{\vec{\mu}_u, S_u\}$ and a vector of gene expression frequencies $\vec{\nu}_u$. If prior cell type information is provided (for example, from Segmentation of cell types (molecule clustering)), the description of each cell is extended to include a cell type label κ_u assigned. Additionally, 'background' is treated as a label and modeled as a separate component (see below). Please notice that here we model background explicitly, which is different from the ideas described in Modeling background. The algorithm also requires prior information about the expected physical cell size s_{global} and variation in cell size σ_{global} with $\sigma_{\text{global}} = s_{\text{global}}/4$ by default. Here, parameters $\vec{\mu}_u$, S_u, $\vec{\nu}_u$ and κ_u are latent variables inferred by the algorithm. It is worth noticing that cell type per molecule cl_i is fixed, while cell type per cell is inferred. So, a cell of type κ_u may include molecules of a type different from κ_u. The graphical model of the process is shown in Extended Data Fig. 1b.

The generative process. The generative process of molecules from a cell (but not background) is defined as

$$P_c(g_i, \vec{x}_i, cl_i | z_i, z_i \neq b, \kappa_{z_i}, \vec{\mu}_{z_i}, S_{z_i}, \vec{\nu}_{z_i}) = \mathcal{N}(\vec{x}_i | \vec{\mu}_{z_i}, S_{z_i}) \cdot \text{Cat}(g_i | \vec{\nu}_{z_i}) \cdot P_{cl}(cl_i | \kappa_{z_i}) \quad (31)$$

Probability $\mathcal{N}(\vec{x}_i | \vec{\mu}_{z_i}, S_{z_i})$ establishes the spatial position and the shape of a cell, $\text{Cat}(g_i | \vec{\nu}_i)$ reflects the expression profile of the cell and probability $P_{cl}(cl_i | \kappa_{z_i})$ penalizes for mismatch between cell-type identity of the cell and a molecule. Mismatch probability is defined as $P_{cl}(cl_i | \kappa_{z_i}) \sim \gamma^{I(cl_i = \kappa_{z_i})}$, with γ set to 0.25 by default.

The generative model for the background component is uniform across all positions, gene and cell-type identity, which effectively assumes constant cell density. For practical purposes, the density p_{bg} of the background component is estimated in such a way so that it would be on the same scale as cell components

$$\begin{aligned} p_{bg} &= f_{bg}^{\text{position}} \cdot f_{bg}^{\text{gene}} \\ f_{bg}^{\text{position}} &= \mathcal{N}\left((3\vec{s}_{\text{global}})_{\text{d}} | \mu = (\vec{0})_{\text{d}}, \Sigma = s_{\text{global}}^2 \cdot I_{\text{d}}\right) \\ f_{bg}^{\text{gene}} &= \frac{1}{n_{\text{comps}}} \sum_{u=1}^{n_{\text{comps}}} \left(\frac{1}{|\{\text{expressed}_u\}|} \sum_{q \in \{\text{expressed}_u\}} \nu_{u,q} \right) \end{aligned} \quad (32)$$

Here, $(\vec{a})_{\text{d}}$ is a constant d-dimensional vector with values a and I_{d} is a d-dimensional identity matrix, $\text{d} \in \{2, 3\}$. And $\{\text{expressed}_u\}$ is the set of all genes with non-zero expression in the component u. The position part of this density corresponds to the level of three expected standard deviations (that is, $3 \cdot s_{\text{global}}$) from the cell center. This part pushes molecules, which are far from any cell, to be assigned to background, and the gene part estimates the average expression probability across all cells and all genes expressed in them (ignoring zeros in the expression probability vectors). This allows for the accounting of sparsity level in the expression vectors.

Prior information. Parameter s_{global} reflects prior knowledge of cell sizes and is described below. In general cases, we aim to incorporate a prior probability $P(S_i | s_{\text{global}}, \sigma_{\text{global}})$ akin to a Wishart prior, where s_{global} and σ_{global} reflect expected size and standard deviation of a cell's diameter. In practice, inference with a covariance prior becomes intractable at scale. We thus use an ad hoc procedure to accommodate prior information, as described below in M-step.

Compared to all segmentation problems described previously, the number n_{comps} of cells, denoted as $\{u_1, .., u_{n_{\text{comps}}}\}$, is unknown in advance and is the subject of statistical inference. This is done by introducing Dirichlet prior $\text{Dir}(\alpha)$ on mixture coefficients $\vec{r}$. Combining the generative model, spatial MRF prior and prior on parameters, the data probability for a cell component is

$$\begin{aligned} &P(g_i, \vec{x}_i, cl_i, z_i, r_{z_i}, S_{z_i} | \kappa_{z_i}, \mu_{z_i}, \vec{\nu}_{z_i}) = \\ &p_{c,i} \cdot P_c(g_i, \vec{x}_i, cl_i | \kappa_{z_i}, \mu_{z_i}, S_{z_i}, \vec{\nu}_{z_i}, z_i) \cdot r_{z_i} \cdot P_{\text{MRF}}(z_i | \vec{z}_{-i}) \cdot \\ &P(S_i | s_{\text{global}}, \sigma_{\text{global}}) \cdot \text{Dir}(r_{z_i} | \alpha) \end{aligned} \quad (33)$$

Deviations from the model. Because the number of cells is large, storing the whole distribution of $Q_i(u)$ becomes the major bottleneck of the algorithm performance in terms of both time and memory. To avoid that, we altered the E-step so that its result is a hard assignment of molecules sampled from the conditional distribution (see below), which corresponds to the stochastic EM algorithm[33]. In the sake of performance, estimation of $Q_i(u)$ was restricted only to components u directly adjacent for the molecule i. Additionally, the Dirichlet process assumes that new components can be added or removed dynamically, which is implemented as separate steps. So, instead of iterating through only E- and M-steps, the workflow is

1. Maximize parameters of the distributions given the existing assignment of molecules by cells (maximization step).
2. Sample empty components from the Dirichlet prior (distribution sampling step).
3. Stochastically assign molecules to components given the exiting components and their parameters (stochastic expectation step).
4. Remove all components that have less than two molecules assigned to them.

After finishing the iterations, the algorithm re-estimates molecule assignment by averaging it over the last N^{est} iterations ($N^{\text{iter}}/10$ by default). These steps are described in more detail below.

E-step. In contrast to the model described in equation (9), the result of this step on the iteration t is a hard assignment of labels $\vec{z}^{(t)}$ but not the distribution $Q^{(t)}(\vec{z})$. So, the MRF probabilities use the indicator function instead of $Q_i(u)$, $P_{\text{MRF}}(z_i) = e^{\beta \cdot \sum_{j \in adj(i)} w_{ij} \mathrm{I}\left[z_j^{(t)} = z_i\right]}$. Then, for a given molecule i, the values $Q_i(u)$ are estimated the usual way; however, it is done only for the components u that are adjacent for this molecule i, $u \in \{z_j\}_{j \in adj(i)}$. The following formulas were used:

$$\begin{aligned} Q_i(u) &\sim p_{c,i} \cdot r_u \cdot P_c(g_i, \vec{x}_i, cl_i | u, \kappa_u, \mu_u, S_u, \vec{\nu}_u) \cdot P_{\text{MRF}}\left(u | \vec{z}_{-i}^{(t)}\right), \; u \neq b \\ Q_i(b) &\sim (1 - p_{c,i}) \cdot p_{bg} \cdot P_{\text{MRF}}\left(b | \vec{z}_{-i}^{(t)}\right) \end{aligned} \quad (34)$$

After estimating the distribution $\vec{Q}_i$, the component label $z_i^{(t+1)}$ is sampled from this distribution.

To reduce the amount of computations, estimation of $Q_i(u)$ was restricted only to components u adjacent to the molecule i on MRF or the background component, which was considered adjacent to all molecules. Additionally, we had to take into account that the distribution sampling step produces components that do not have any molecules assigned to them. For every component u that was just sampled from the prior and still has no assigned molecules, the algorithm found a molecule i closest to the component center $\vec{\mu}_u$ and

included component k as adjacent for all molecules j that are connected to i: $\left(i = \text{argmin}\, l \in 1 : n_{\text{mols}}(|\overrightarrow{x_i} - \overrightarrow{\mu_u}|)\right) \Rightarrow (u \in adj(j),\ \forall j \in \cup(adj(i), i))$.

M-step. For the categorical distribution of the component k, non-normalized probability $v_{k,q}$ of the gene q being expressed was estimated as a fraction of the gene q across the observed molecules (smoothed with pseudocounts; see M-step):

$$v_{k,q} = \frac{|\{(g_i = q)\&(z_i = k)\}|_{i \in 1:n_{\text{mols}}} + 1}{|\{z_i = k\}|_{i \in 1:n_{\text{mols}}} + 1} \tag{35}$$

For the multivariate normal distribution, the mean $\overrightarrow{\mu_k}$ was estimated as the average over the positions of the assigned molecules:

$$\mu_{k,u} = \frac{1}{n_k} \sum_{i:\, z_i = k} x_{i,u}, \tag{36}$$

Here, $u \in \{1,2\}$ or $\{1,2,3\}$ and $n_k = |\{z_i = k\}|_{i \in 1:n_{\text{mols}}}$ is the total number of molecules assigned to the component k. The maximal likelihood estimator was used for the covariance matrix:

$$S_k = \frac{1}{n_k} \sum_{i:\, z_i = k} \left((\overrightarrow{x}_i - \overrightarrow{\mu}_k) \cdot (\overrightarrow{x}_i - \overrightarrow{\mu}_k)^T \right) \tag{37}$$

After estimating the covariance matrix, it was adjusted based on the global scale s_{global}. The most popular solution for such adjustment uses the Wishart prior over the covariance matrix, as this prior is conjugate for the normal distribution. However, the Wishart prior is parameterized with the expected covariance matrix and the number of degrees of freedom and thus does not allow us to explicitly control the magnitude of deviation from the expected covariate matrix. Therefore, we instead relied on the non-conjugate normal-like prior on the eigenvalues of the covariance matrix. This prior was parameterized with the expected size of the eigenvalues s_{global} and their standard deviation σ_{global}, which can be specified by the user or set to $0.25 \cdot s_{\text{global}}$ by default. Then, the adjustment starts by performing eigen decomposition over S_k to calculate its eigenvalues $\overrightarrow{\lambda}_k$ and the eigenvector matrix Q_k. Next, the eigenvalues are adjusted using the formula $\lambda'_{k,u} = \lambda_{k,u} + \text{sign}(\lambda_{k,u} - s_{\text{global}}) \cdot \sqrt{\frac{|\lambda_{k,u} - s_{\text{global}}|}{\sigma_{\text{global}}}} \cdot \sigma_{\text{global}}$, where $u \in [1,2]$. This transformation corresponds to quadratic penalty over deviation z scores $Z_{k,u}$, reducing the deviation $Z_{k,u} \cdot \sigma_{\text{global}}$ to $\sqrt{Z_{k,u}} \cdot \sigma_{\text{global}}$. Next, to account for the components with a low number of samples, it is further adjusted as $\lambda_{k,u}{}'' = \frac{m \cdot \sigma_{global} + n_k \cdot \lambda'_{k,u}}{m + n_k}$, where m is the expected minimal number of molecules per cell and n_k is the number of molecules assigned to the component k. Finally, the adjusted covariance matrix is estimated as $S'_k = Q_k \Lambda'' Q_k^{-1}$, where Λ'' is the diagonal matrix with $(\lambda_{k,1}'')^2$ values on the diagonal.

Mixture coefficients $\overrightarrow{r}$ are estimated by adjusting proportions of molecules, assigned to each component, by the Dirichlet prior. However, for the Dirichlet process to have more global convergence, instead of using exact posterior estimates for $\overrightarrow{r}$, these values were sampled from the posterior after each iteration

$$\overrightarrow{r} \sim \text{Dir}\left(\max(\overrightarrow{n}, \alpha)\right) \tag{38}$$

Here, α is the Dirichlet Process parameter set to 0.2 by default.

If the prior cell type segmentation is available, the parameter κ_u is estimated as the mode of cell-type labels cl_i across all molecules i assigned to the cell u

$$\kappa_u = \text{mode}(\{cl_i\}_{i:z_i = u}) \tag{39}$$

Distribution sampling. To allow arbitrary numbers of components in the data, we used a dynamic version of the truncated Dirichlet prior[34]. For that, after the maximization step, $n_{\text{nc}} = \rho_{\text{new}} \cdot n_{\text{comps}}$ new components were sampled from the prior distributions. The parameter ρ_{new} was set to 0.3 by default. For the normal distribution, centers were sampled from all molecule positions with the weights proportional to the molecule confidences, $\mu^* = \overrightarrow{x}_i; i \sim \text{Cat}(\overrightarrow{p}_c)$. The diagonal covariance matrix S^* was sampled from the global scale prior with diagonal values $s_i \sim \mathcal{N}\left(s_{\text{global}}, \sigma^2_{\text{global}}\right)$, $i \in \{1,2\}$ or $\{1,2,3\}$ depending on the data dimensionality. Sampling gene composition parameters requires proper modeling of sparsity of the expression vectors, which varies greatly between protocols and cell types. It is therefore unclear how to capture this with a parametric prior distribution. Instead, the algorithm sampled gene composition parameters uniformly from the existing components and randomly permuted the vector $\overrightarrow{v}^* = \text{shuffle}(v_k)$; $k \sim \text{Uniform}(1{:}\, n_{\text{comps}})$.

Using compartment assignment. Information about compartment assignment per molecule or information about some genes being compartment specific is sometimes available. This can be used to improve the quality of the segmentation in several ways. First, knowing which molecules belong to the nuclei and which came from the cytoplasm can improve the inference of cell center positions. Second, it can also help to separate cells in dense regions, where it is challenging to define exact boundaries in the cytoplasm but nuclei are somewhat separated from each other. Third, by requiring each cell to have some nuclei-specific molecules, the algorithm can avoid oversegmentation when some cells can be assigned to purely cytoplasmic regions. This issue is particularly noticeable when the expression patterns vary across compartments, which violates our assumption on homogeneity of the transcriptional composition within a cell, as a result pushing the algorithm to assign each compartment to a separate component. This problem can be mitigated by knowing which genes have such compartmentalized expression and treating them differently.

When a list of compartment-specific genes is provided, such information can be inferred as described in Compartment segmentation. In this case, the probability that a molecule i comes from a nucleus is estimated as $p_{n,i} = 1 - {}_{pi}(\text{'cyto'})$ so that all molecules in 'na' regions are assumed to come from nuclei. Then, the M-step is adjusted, using weighted estimators for the cell center positions $\overrightarrow{\mu}_k$

$$\mu_{k,u} = \frac{\sum_{i:z_i = k} p_{n,i} \cdot x_{i,u}}{\sum_{j:z_j = k} p_{n,j}} \tag{40}$$

Additionally, a cell is penalized if it does not have enough molecules with high $p_{n,i}$. For that, during the M-step, the penalty factor τ_k for a cell k is estimated as 0.9th quantile of $p_{n,i}$ across all molecules assigned to k. Then, during the E-step, assignment probability $Q_i(k)$ is multiplied by τ_k. The raw probabilities ${}_{pi}(\text{'cyto'})$ and ${}_{pi}(\text{'nuclei'})$ are also used to adjust the MRF weights so that nuclei molecules can spread their labels to cytoplasm but not the other way around. To do so, the MRF is updated as $w'_{i,j} = w_{i,j} \cdot [1 - {}_{pi}(\text{'cyto'}) \cdot {}_{pi}(\text{'nuclei'})]$ for all i with ${}_{pi}(\text{'nuclei'}) > 0.25$. Finally, to deal with the negative influence of compartment-specific genes, gene identity of all molecules coming from those genes are set to *NA* before starting the segmentation optimization. Then, during the E-step, the categorical part of the density equation (34) for such molecules is omitted

$$Q_i(u) \sim p_{c,i} \cdot r_u \cdot \mathcal{N}(\overrightarrow{x}_i | \overrightarrow{\mu}_{z_i}, S_{z_i}) \cdot P_{cl}(cl_i | \kappa_{z_i}) \cdot P_{\text{MRF}}\left(u | \overrightarrow{z}^{(t)}_{-i}\right) \tag{41}$$

Using a prior segmentation. For many datasets, auxiliary microscopy stains, such as DAPI, can be used to generate a prior segmentation. Such a segmentation can be used to resolve the cases where measured transcript molecules do not provide sufficient information. The complexities of segmenting and aligning the auxiliary stains, however, mean that the quality of such segmentations can vary. To incorporate this optional information, Baysor can accept prior cell segmentation labels per molecule as well as the prior segmentation confidence parameter $c_{\text{prior}} \in [0;1]$. At $c_{\text{prior}} = 0$, the prior segmentation would be ignored, whereas $c_{\text{prior}} = 1$ forces Baysor segmentation not to violate the prior segmentation assignments. Increasing c_{prior} from 0 to 1 gradually increases the importance of the prior segmentation. The prior segmentation penalty is evaluated only for the molecules that are assigned to some cell (but not to background) in both Baysor and the prior segmentations. This accounts for the fact that the imaging-based segmentations may miss some cells or portions of cells that can still be deduced from the spatial transcriptomics data. The most obvious example of such a situation is the DAPI-based segmentations, which cover only molecules within the cell nuclei, leaving most of the cytoplasm molecules unannotated. The Baysor algorithm, therefore, treats background labels (that is, not within the segmented region) within the prior segmentations as 'unknown' instead of explicitly assigning these molecules to the background component. The situation in which some non-background molecule from the prior segmentation is recognized as background by Baysor is dealt with during Separation of the intracellular molecules from the background. Specifically, if a molecule i is recognized as intracellular in the prior, its confidence cannot be less than c^2_{prior}: $p_{c,i} \geq c^2_{\text{prior}}, \forall i : z_i^{\text{prior}} \neq \text{background}$. As a result, there are two possible types of contradictions between the two segmentations: (1) multiple Baysor components are present within one prior segment and (2) one Baysor cell component touches multiple prior segments.

The penalties are applied during the stage at which the densities of different components are estimated for a given molecule by multiplying $Q_i(z_i)$ by the penalty term $\beta_{k,i}$. If this molecule has a prior assignment to some segment, all but the component with the largest intersection with this prior segment are penalized. This is done using the following procedure. After the distribution sampling step, the algorithm estimates the number of molecules $n^{\text{seg}}_{k,q}$ per segment q for each of the cell components k. Then, based on these numbers, it estimates the main prior segment per component ς_k. This main segment label is used to separate the cases where two Baysor components are present within one prior segment (in this case, they would have the same main segment ID) from the cases where a Baysor component touches multiple segments (it would have one main segment but would get penalized for touching all other segments). During the expectation step for molecule i, the algorithm finds the component u^*, which has the largest intersection with the segment $z_i^{\text{prior}} = q$ across all candidate components for any molecule i that has this segment as their main assignment or does not have a main segment at all, $u^* = \text{argmin}\, k : \varsigma_k \in (q, \varnothing)\left(n^{\text{seg}}_{k,q}\right)$. If any two components have the same number of molecules for this segment, the component with the larger total number of molecules is chosen as the main component. The main component

per segment incurs no penalty ($\beta_{u^*,i} = 1$), while the other components are penalized as

$$\beta_{k,i} = \begin{cases} \sqrt{1 - c_{\text{prior}}} \left(\frac{n^{\text{seg}}_{k,q}}{n^{\text{seg}}_{u^*,q}} \right)^{c_{\text{prior}}} & \text{if } \varsigma_k \in (q, \varnothing) \\ \sqrt{1 - c_{\text{prior}}} \left(\frac{n^{\text{seg}}_{k,q}}{n^{\text{seg}}_{q}} \right)^{c_{\text{prior}} \cdot \exp(3 \cdot c_{\text{prior}})} & \text{otherwise} \end{cases} \quad (42)$$

Here, n^{seq}_q is the total number of molecules per segment q. The first penalty line prevents two cells from existing in the same prior segment (oversegmentation problem), while the second line prevents one cell from taking over a big part of a segment assigned to a different cell (overlapping problem). The second penalty line also solves the undersegmentation problem; if a Baysor cell overlaps two prior segments, then as soon as a new component is sampled inside one of these segments, the existing 'doublet' cell incurs a very large penalty for all other molecules of this segment. These functional forms of the penalties, while somewhat arbitrary, have allowed us to express a desired functional shape for the penalties (Supplementary Fig. 10b,c). The penalties are also scaled in [0; 1], which allows us to avoid changes to the distribution sampling step. Newly sampled cells have no penalty, which corresponds to $\beta_{k,i} = 1$.

Initialization. The algorithm is initialized with a large number of components uniformly distributed over 2D space. Setting the initial number of cells to be much larger than the expected number greatly improved the convergence of the algorithm. The initial cell centers were selected uniformly across 2D space, and each molecule was assigned to the nearest center. The center selection was done using a strategy similar to the one used for subsampling of the NCVs; the molecules were ranked by the sum over the x and y coordinates, and then the algorithm selected a subset from this array uniformly across the ranks. When molecule background confidences were available, only molecules with a true signal confidence greater than 0.25 were used. The MRF strength parameter beta was set to 0.1 for all experiments.

Adjusting MRF. The MRF was initialized using Delaunay triangulation for 2D space and five NN graphs for 3D space, as described in Building the random field. However, given that the triangulation uses only the information about the spatial positions, the MRF was then further adjusted to reflect the neighborhood composition similarities based on NCVs. Specifically, NCVs were estimated for each molecule, and the resulting matrix of NCVs was transformed using PCA. Pairwise Pearson linear correlations of the PC vectors, $\rho_{i,j}$, were estimated for any two molecules i and j that were connected by an edge in the graph, and the MRF edge weight was then multiplied by $\max(\rho_{i,j}, 0.01)$. When information about compartment assignment was available, the MRF was also adjusted as described in Using compartment assignment.

Stop criteria. Given the sampling nature of the algorithm, estimating convergence is challenging. Instead, the algorithm always works for a fixed number of iterations (500 by default, which was well sufficient for the algorithm to converge on a fixed set of cells and their parameters).

Complete Baysor workflow. Below is the description of the full Baysor workflow, as implemented by the command line interface:

1. Read the data frame with information about molecules. Filter out the genes with the total number of molecules below threshold if it was specified.
2. If provided, load the prior segmentation mask. Filter out any prior segments that have less than m^{prior} molecules. Estimate the global scale s^{global} based on these data.
3. Estimate confidence $p_{c,i}$ per molecule.
4. If a list of compartment-specific genes is provided, estimate compartment assignment per molecule.
5. If the specified number of cell clusters (four by default) is greater than one, run the cell-type segmentation using the confidences estimated above. Otherwise, assign all molecules to the same cluster.
6. If a list of compartment-specific genes is provided, set all g_i for the molecules produced by these genes to *NA*.
7. Initialize the BMM algorithm.
8. Run the BMM algorithm.
9. Re-estimate assignment by averaging over the last N^{rest} iterations.
10. Save the segmentation results.
11. Build diagnostic plots.

Inferring the algorithm parameters. Baysor implementation derives most of the parameter estimates based on the minimal expected number of molecules per cell m, which must be specified by the user, as well as the number of genes measured by the assay (n_{genes}). Parameter m depends on the protocol and the measurement. This key parameter can impact segmentation performance (Supplementary Fig. 4), in particular as a result of the molecule background probability determination step. It is therefore recommended for the users to check diagnostic plots for the background assignment step to make sure the value of m is suitable. With larger values of m, the algorithm will expect more molecules concentrated in the same region to be recognized as a cell. Other parameters are inferred as

- The initial number of cells is set to $N^{\text{init}}_{\text{cells}} = N_{\text{molecules}}/m$, where $N_{\text{molecules}}$ is the total number of molecules.
- The number of PCs for the PCA transformation of NCVs for adjusting the MRF is estimated as $\min\left(\max\left(n_{\text{genes}}/3, 30\right), 100, n_{\text{genes}}\right)$.
- The NN index for estimating distances d_i during the background segmentation is set to $\max(m/2 + 1, 2)$.
- The number of NNs for NCV coloring is set to $\max(n_{\text{genes}}/10), m, 3$.
- If a segmentation mask was provided, it is used to infer the global scale s_{global} and its standard deviation σ_{global}. To do so, Baysor first approximates cell radii from the number of pixels per prior segment n^{pix}_i as $r_i = n^{\text{pix}}_i/\pi$. Then, $s_{\text{global}} = \text{median}_i(r_i)$ and $\sigma_{\text{global}} = 1.4826 \cdot \text{MAD}_i(r_i)$, which are the median and the adjusted median absolute deviation (MAD) over the radii.
- Without a segmentation mask, the global scale s_{global} is a required input parameter, and the deviation σ_{global} is set to $0.25 \cdots_{\text{global}}$ by default.
- The minimal number of molecules per prior segment $m^{\text{prior}} = \max(m/4, 2)$.

Segmentation parameters. Parameters of the segmentation runs for different datasets are shown in the Supplementary Table 3.

Benchmarks. To evaluate the performance of Baysor and compare it with other methods, we have used several benchmarks that measure the agreement with segmentations based on poly(A) signal, outer membrane immunostaining and examination of the differences between segmentations based on transcriptional correlation of discordant regions.

Poly(A) benchmarks. The MERFISH dataset was used, taking poly(A) segmentation from the original publication[19]. The segmentations were obtained by Baysor, pciSeq and DAPI Watershed algorithms (see the corresponding sections below). For each poly(A) (Source) cell, we estimated a cell with the largest overlap from each of the Baysor/pciSeq/Watershed cells (Target). For each of the Target cells, Source segmentation was considered as the ground truth, and two metrics were estimated. The first metric is the number of molecules in the overlap of the Source and the Target cell divided by the number of molecules in the Source cell. This ratio corresponds to recall, a common machine learning quality metric. The second metric is the overlap size divided by the number of molecules in the Target cells, which corresponds to precision.

Correlation benchmarks. As it is difficult to establish ground truth on cell segmentations, we developed a benchmark to evaluate the differences between two segmentations. For each Source cell c in the segmentation $\vec{z}_{\text{src}}$, the benchmark finds a Target cell t^* from the segmentation $\vec{z}_{\text{target}}$ that has the largest overlap with c (Fig. 4c). Then, it splits the molecules from c over the part that overlaps with t^* from the molecules in the non-overlapping part. Measuring similarities of gene compositions of these two parts as a Pearson linear correlation (ρ_c), the benchmark evaluates the distribution of such similarities over all cells from the Source segmentation, $\vec{\rho} = \{\rho_c\}_{c \in \vec{z}_{\text{src}}}$. Taking two segmentations $\vec{z}_1$ and $\vec{z}_2$ as the input, the benchmark returns two distributions $\vec{\rho}_1$ and $\vec{\rho}_2$ for the cases where $\vec{z}_{\text{src}} = \vec{z}_1$, $\vec{z}_{\text{target}} = \vec{z}_2$ and $\vec{z}_{\text{src}} = \vec{z}_2$, $\vec{z}_{\text{target}} = \vec{z}_1$ correspondingly. Based on these two distributions, the segmentation is said to be better if it has higher $\vec{\rho}_i$ values. More formally, given two segmentations $\vec{z}_{\text{src}}$ and $\vec{z}_{\text{target}}$, the benchmark procedure does the following:

1. All cells i with the total number of molecules below the threshold ($n^{\text{mols}}_i < m$) are removed from both segmentations.
2. Among the remaining cells, a contingency matrix between the two segmentation assignments is estimated.
3. Based on the contingency matrix, for each cell c of the segmentation $\vec{z}_{\text{src}}$ and cell t from the segmentation $\vec{z}_{\text{target}}$, the overlap fraction $f^{\text{over}}_{c,t}$ is estimated.
4. For each cell $c \in \vec{z}_{\text{src}}$, the target cell $t^* \in \vec{z}_{\text{target}}$ is determined as the cell with the highest molecule overlap fraction $f^{\text{over}}_{c,t^*} = \frac{|\{i : (z_{\text{src},i} = c) \;\&\; (z_{\text{target},i} = t^*)\}_{i \in 1:N^{\text{mols}}}|}{n^{\text{mols}}_c}$.
5. Only cells from $\vec{z}_{\text{src}}$ with the overlap $f^{\text{over}}_{c,t^*} \in (25\%, 75\%)$ are selected for further analysis.
6. For each cell $c \in \vec{z}_{\text{src}}$, the molecules from this cell are partitioned into (1) the main overlap $\vec{l}^{\text{main}}_{c,t^*} = \{i : (z_{\text{src},i} = c) \;\&\; (z_{\text{target},i} = t^*)\}$ and (2) the rest $\vec{l}^{\text{rest}}_{c,t^*} = \{i : (z_{\text{src},i} = c) \;\&\; (z_{\text{target},i} \neq t^*)\}$.
7. Gene composition vectors $\vec{v}^{\text{main}}_{c,t^*}$ and $\vec{v}^{\text{rest}}_{c,t^*}$ are estimated over the molecules from $\vec{l}^{\text{main}}_{c,t^*}$ and $\vec{l}^{\text{rest}}_{c,t^*}$ correspondingly.
8. Pearson linear correlation coefficients ρ_c between the vectors $\vec{v}^{\text{main}}_{c,t^*}$ and $\vec{v}^{\text{rest}}_{c,t^*}$ are estimated. If these two sets of molecules were produced by different cell types, the correlation ρ_c is expected to be low, suggesting that the segmen-

tation $\vec{z}_{src}$ was erroneous for the cell c. However, similar to the classical hypothesis testing framework, a high correlation ρ_c does not mean that the segmentation $\vec{z}_{src}$ is correct; it only suggests that the molecules were obtained from the same cell type, and it is still possible that they originated from different cells.

Given that assessment of such expression correlation between parts of the cell requires a relatively high number of molecules per cell, we were not able to apply this benchmark to the dataset generated using the ISS protocol[16] (Extended Data Fig. 4a). Also, the STARmap datasets[18] do not have published auxiliary stainings, so we were unable to run Watershed and pciSeq segmentations on them. As a result, for comparison with the Watershed and pciSeq data, which require nuclei stains, STARmap was excluded as well.

Sensitivity benchmarks. To assess the stability and sensitivity of the algorithm, Baysor was run seven times with different random number generator seeds on a subset of the MERFISH hypothalamus dataset ($x \in [6{,}000, 10{,}000]$, $y \in [6{,}000, 10{,}000]$ in pixel coordinates). Each run used 400 iterations with parameters 'scale=6.5' and 'min_molecules_per_cell=30', 'new_component_frac=0.3', 'new_component_weight=0.2'. The final assignment was obtained by averaging over the last 100 iterations ('assignment_history_depth=100'). Then, parameters were varied one by one (Supplementary Fig. 4), and mutual information was estimated for all pairs of the baseline segmentations and the segmentations with changed parameters. Additionally, sensitivity to varying sparsity of the dataset was estimated. For that, the dataset was split over 25 large squares, and those with even IDs were progressively downsampled from 100% to 10% of their original number. Then, the segmentation was run on the downsampled dataset, and mutual information between the assignments was estimated separately for the molecules in the downsampled and preserved regions.

Performance benchmarks. To assess the runtime performance profiling of molecule clustering (Supplementary Table 1), the MERFISH dataset was used with the molecules subset to those with coordinates $x < -3{,}300$ and $y < -3{,}300$ (338,023 molecules in total). Molecule confidence was then estimated using the confidence_nn_id = 26 parameter setting. The MRF clustering was run five times for each value of $k \in \{2, 4, 6, 8, 10\}$, with parameters max_iters = 5,000 and n_iters_without_update = 100.

To evaluate the performance of the NCV Leiden clustering, the Pagoda2 package was used. For each run, the NCV matrix was estimated using 50 NNs normalized by the total count of each NCV and reduced to projections on the top 50 PCs. A k-NN graph was then built using 30 NNs (using the default cosine similarity distance metric), and the Leiden community detection algorithm was applied with the default resolution = 1.0. For the profiling, the same parameters as specified in Supplementary Table 3 (prior = no) were used. Each dataset was segmented five times using a single thread. All benchmarks were run on a Lenovo Thinkpad X1 laptop with Intel(R) Core(TM) i7-8850H at 2.60 GHz CPU and 64 GB RAM.

Quality metrics. After finishing the segmentation, Baysor extracts the following summary metrics per cell:

- Cell area: area of the convex hull around molecules assigned to the cell.
- Density: a cell's area divided by the number of molecules in the cell.
- Elongation: ratio of the two eigenvalues of the cell shape covariance matrix.
- Average confidence: background/signal confidence averaged across all the molecules assigned to the cell.

scRNA-seq processing. *Analyzing NCVs using scRNA-seq tools.* To generate clustering and embedding of NCVs (Fig. 1b,c), the Pagoda2 package (https://github.com/kharchenkolab/pagoda2/) was used. The data were preprocessed using total count normalization, and then a 50-NN graph over the normalized counts using the default cosine distance metric was built. Leiden clustering with parameters 'resolution=8' and 'n.iterations=15' was used for annotation expansion (see below). Visualization of the MERFISH and the osmFISH datasets was done by building a joint matrix over the paper and the Baysor segmentations, reducing its dimensionality with PCA to the 10 top PCs (MultivariateStats.jl package) and then running UMAP embedding (UMAP.jl package) using Euclidean distance and parameters 'spread=2.0' and 'min_dist=0.1' over the joint matrix. Hierarchical clustering with the Ward linkage method and 70 clusters (Clustering.jl package) was used for the annotation expansion.

scRNA-seq annotation. The annotation for Allen smFISH data was generated using the CellAnnotatoR package. The cell-type markers were inferred from mouse VISp scRNA-seq data produced for the SpaceTx consortium by the Allen Brain Institute (personal communication) and then applied to the corresponding FISH data. For inference, the 'merged_cluster_smFISH' annotation shared with the dataset was used. First, the cell-type hierarchy was built using 'broad_class' as the first level, class prefix from 'merged_cluster_smFISH' as the second level and the full 'merged_cluster_smFISH' labels as the third level of the hierarchy. Second, cell types 'CR', 'Astro', 'Endo', 'Macrophage', 'Oligo' and 'SMC' were removed, as they could not be distinguished by the genes measured in the Allen smFISH dataset. Next, a CellAnnotatoR marker inference procedure was used to obtain the markers using this hierarchy, and the resulting marker list was adjusted by hand to improve the quality of the annotation. Finally, CellAnnotatoR was used to apply the identified markers to the NCV data. To annotate MERFISH and osmFISH data, the marker list was compiled manually using the information published by the protocol authors, and then CellAnnotatoR was applied in the same manner. In all three cases, the resulting annotation was expanded over the clusters found in the previous step (see Analyzing NCVs using scRNA-seq tools).

Polygon visualization. To visualize the boundaries of the cells, a kernel density estimation (KDE)-based algorithm was used. The algorithm builds a grid over 2D space, assigns each grid node to the cell with the highest density of molecules around this node and draws a polygon around this label on the grid. More formally, the following steps were performed:

1. A uniform four-connected grid was created over the 2D space, with the grid step specified as an input parameter.
2. The density of molecules for each cell was estimated over the grid nodes using the KDE implementation in the KernelDensity.jl package. KDE bandwidth, a parameter of the algorithm, was set to the 0.5 * 'grid step' by default. To improve runtime performance, only the nodes within three bandwidths of the cell molecules were taken into account.
3. Each node was assigned to the cell with the maximal density in this node. If the maximal density was below threshold (10^{-5} by default), the node was assigned to the background.
4. For each cell, the graph of boundary nodes was determined as the grid nodes of the cell that were adjacent to the nodes from the other cells or from the background. The edges between these boundary nodes were accepted into the boundary graph.
5. For each cell, a minimal spanning tree was built over its boundary graph using the Kruskal algorithm.
6. For each cell, the longest path in this tree was extracted using the Dijkstra algorithm, and the resulting path was transformed into a polygon by connecting its beginning and its end.

Baysor segmentation. All datasets were segmented using the Baysor command line interface, with parameters specified in the Supplementary Table 3.

pciSeq segmentation. All datasets were segmented using the pciSeq Python package v0.0.30 with default parameters. As input, we passed the spot matrix, Watershed-segmented DAPI stains and clustered scRNA-seq data. The following processing steps of scRNA-seq data were used:

- Allen smFISH: the annotated mouse VISp scRNA-seq data produced for the SpaceTx consortium by the Allen Brain Institute (personal communication) was used.
- MERFISH hypothalamus data: the dataset from the original publication[19] was processed using the scanPy package[35]. Total-count and log-normalization were applied, then highly variable genes were selected, and gene variance scaling was applied with subsequent PCA dimensionality reduction. Finally, a k-NN graph was built using 15 PCs and $k = 30$. Leiden clustering with resolution = 3 was performed on this graph, resulting in 44 clusters. These clusters were used as cell types for the pciSeq run.
- osmFISH: the published mouse somatosensory cortex scRNA-seq dataset[36] was used with the published annotation (level 2). Two genes from the FISH data were missing in the original dataset, likely due to typos in their names; so to fix it, we renamed '*Tmem2*' to '*Tmem6*' and '*Kcnip2*' to '*Kcnip*' in the scRNA-seq data.
- ISS: the dataset produced by Harris, et al.[37] suggested by the original ISS publication[16] was used with the published annotation.
- STARmap datasets do not have any stainings published, so they were excluded from the comparison.

After obtaining scRNA-seq data with cell-type assignment, counts were summed across all cells for each of the cell types, and the aggregated count matrices were passed to pciSeq. An exception to this workflow is the MERFISH mouse gut dataset, which has 3D DAPI stacks with no scRNA-seq data available. So, Cellpose[27] DAPI segmentation for $z = 8.5$ (stack ID = 5) was used for the whole 3D dataset. Instead of scRNA-seq data, we used transcriptional count (MERFISH) profiles of cells obtained using 3D Watershed segmentation of the membrane IF signal. They were processed with the scanPy package using total count normalization and PCA reduction to 15 components. Then, a k-NN graph was built ($k = 30$) and clustered with the Leiden algorithm (resolution = 3).

DAPI and membrane segmentation with Cellpose. On the MERFISH mouse gut dataset, the quality of the watershed segmentation was too low due to the cell density, so we used the Cellpose[27] software fine-tuned for our data. For DAPI stains, we selected one field of view and manually labeled all nuclei on it for one z slice ($z = 8.5\,\mu m$). Then, the Cellpose nuclei model was fine-tuned (with the default parameters) using these manually labeled data. Finally, it was applied for all z slices

over the full dataset with diameter = 120. For membrane stains, we selected three different fields of view and manually segmented one z slice from each of them (z = (4, 8.5, 14.5) µm). Then, we fine-tuned the cyto Cellpose model (using default parameters) and applied it to the whole dataset with diameter = 60. To construct 3D cell boundaries, Cellpose cell boundaries defined by DAPI or membrane stain were stitched together across z stacks so that the same cell has the same ID across z stacks by matching cells between adjacent z planes. Specifically, cell b in plane 2 is considered to be the same cell as cell a in plane 1 if cell a overlaps with the shadow of cell b in plane 1 by over 60% OR if cell b's shadow overlaps predominantly with the blank space in plane 1, yet cell a occupies over 15% of cell b's shadow in plane 1 AND among all the cells in plane 1 that overlap with cell b, cell a accounts for over 90% of the overlapping area.

DAPI Watershed segmentation with ImageJ. To generate prior segmentations using DAPI stains, the Watershed segmentation was performed using ImageJ[24] software. Each staining was segmented by the following procedure:

1. The image was converted to 8-bit format ('Image'/'Type'/'8-bit').
2. A median filter with 1 pixel radius was applied ('Process'/'Filters'/'Median...').
3. Auto Threshold with the 'Default' method and 'Ignore black' option was applied for image binarization ('Image'/'Adjust'/'Auto Threshold').
4. Watershed segmentation was applied ('Process'/'Binary'/'Watershed').

VISp multiplexed smFISH data generation. Multiplexed smFISH data of mouse primary visual cortex (VISp) was generated as part of the SpaceTx consortium. Tissue processing was performed as previously described[38], with some modifications. The description is taken from ref. [21].

Silanization of coverslips (1.5, Thorlabs CG15KH) was performed by plasma cleaning for 30 min in Plasma-Prep III (SPI 11050-AB) followed by vapor deposition of 3-aminopropyl-triethoxysilane (APES; Sigma A3648) in a vacuum for 10 min. Coverslips were then washed in 100% methanol for 2 × 5 min, allowed to dry and stored in a dust-free environment until use.

Fresh–frozen mouse brain tissue was sectioned at 10 µm onto silanized coverslips, allowed to dry for 20 min at –20 °C then fixed for 15 min at 4 °C in 4% paraformaldehyde (PFA) in PBS. Sections were washed 3 × 10 min in PBS and permeabilized and dehydrated with chilled 100% methanol at –20 °C for 10 min and allowed to dry. Sections were stored at –80 °C until use. Frozen sections were rehydrated in 2× SSC (Sigma 20XSSC, 15557036) for 5 min and treated 10 min with 8% SDS (Sigma, 724255) in PBS at room temperature. Sections were washed five times in 2× SSC. Sections were then incubated in hybridization buffer (10% formamide (vol/vol); Sigma, 4650), 10% dextran sulfate (wt/vol) (Sigma, D8906), 200 µg ml^{-1} bovine serum albumin (BSA; ThermoFisher, AM2616), 2 mM RVC (New England Biolabs, S1402S) and 1 mg ml^{-1} tRNA (Sigma, 10109541001) in 2× SSC for 5 min at 37 °C. Probes were diluted in hybridization buffer at a concentration of 250 nM and hybridized at 37 °C for 2 h. Following hybridization, sections were washed 2 × 10 min at 37 °C in wash buffer (2× SSC, 20% formamide), 1 × 10 min in wash buffer with 5 µg ml^{-1} DAPI (Sigma, 32670) and three times with 2× SSC. Sections were then imaged in imaging buffer (20 mM Tris-HCl, pH 8, 50 mM NaCl, 0.8% glucose (Sigma, G8270), 30 U ml^{-1} pyranose oxidase (Sigma, P4234), 50 µg ml^{-1} catalase (Abcam, ab219092)). Following imaging, sections were incubated 3 × 10 min in stripping buffer (65% formamide, 2× SSC) at 30 °C to remove hybridization probes from the first round. Sections were then washed in 2× SSC for 3 × 5 min at room temperature before repeating the hybridization procedure.

The multiplexed smFISH image data were collected and processed using methods previously described[38], except that the images from different rounds of hybridization were registered in (x,y) based on the DAPI signal. The spot locations and raw data are available on request.

MERFISH measurements in mouse gut. *MERFISH encoding probe design and construction.* In total, 241 genes were targeted for MERFISH measurements in the mouse small intestine. To construct an encoding probe set for these genes, we first created a binary encoding scheme with a minimum Hamming distance between all barcodes of 4 and a constant Hamming weight of 4. This 20-bit barcoding scheme contains 256 barcodes, of which 15 were left unassigned to serve as measures of the false-positive rate in these measurements.

Marker gene selection for major cell classes in the mouse small intestine was guided using previously published scRNA-seq measurements[39,40], and isoform selection was guided using published bulk RNA-seq data for the mouse small intestine[41]. In many cases, we targeted a single isoform of a given gene; however, in some cases, we designed encoding probes to target shared regions of multiple isoforms of individual genes. Barcodes were assigned to each of these genes at random.

Templates for the production of MERFISH encoding probes were designed using a previously published[42] probe design pipeline using standard parameters for the mouse genome: 30-nucleotide regions of homology to the target regions, melting temperatures between 65 °C and 75 °C, GC contents between 0.4 and 0.6, gene specificities between 0.75 and 1 and isoform specificities between 0.75 and 1 (where appropriate). Designed target regions were allowed to overlap with other potential target regions by as much as 20 nucleotides. Seventy-two encoding probes were designed for each gene with each encoding probe containing three of the four readout sequences associated with the barcode assigned to each gene. In rare instances (20 of 241 genes), some genes were too short to support 72 encoding probes; however, only two genes had fewer than 40 encoding probes, and no gene had fewer than 31 encoding probes.

Encoding probe templates were synthesized as a complex oligopool (Twist Biosciences). This pool was then used to produce encoding probes via published methods[42]. Briefly, an in vitro transcription template was made from the oligopool via limited-cycle PCR and purified via spin column (Zymo Spin V with the DNA Clean and Concentrator protocol; C1012-50). RNA was created from this template using high-yield in vitro transcription (NEB HiScribe, E2050S) and purified with solid-phase reversible immobilization beads (SPRI). DNA was produced from this RNA using reverse transcription (Thermo, Maxima RT H minus). The RNA template was removed via alkaline hydrolysis, and the single-stranded DNA probes were purified with SPRI beads.

Oligo-labeled secondary antibodies. Oligo-labeled goat anti-rabbit secondary antibodies were produced as described previously[25]. Briefly, secondary antibodies (goat anti-rabbit IgG, 2 mg ml^{-1}; Thermo, A16112) were purified via three 1× PBS washes in a size-exclusion spin column (Amicon Ultra-0.5 Centrifuge Filters, EMD, UFC510024) and then exposed to 100 µM of a DBCO-NHS crosslinker (Kerafast, FCC310) for 1 h at room temperature. Unreacted crosslinker was then removed by three additional 1× PBS washes with the same size-exclusion column, and an azide and acrydite-modified oligo (/5Acryd/GAGGTAGGGAGTATGTAGTT/3AzideN/; IDT) was added to a final concentration of 20 µM. Antibody–oligo mixes were incubated overnight at 4 °C to allow the DBCO-azide click reaction. Excess oligo was not purified from antibodies.

Mouse ileum collection and cryosection. Male-specific pathogen-free C57BL/6 mice aged 8 to 12 weeks were killed via isoflurane anesthesia followed by cervical dislocation using a protocol approved by the Harvard Medical School Office of the IACUC. The small intestine was rapidly dissected and dropped into ice-cold 4 mM RVC (NEB, S1402S) in 1× PBS in a large Petri dish on ice. The ileum was then rapidly excised, cut into two halves and carefully flushed with cold 4 mM RVC. The tissue was left to incubate in 4 mM RVC at 4 °C for 1 h before embedding in prechilled optimal cutting temperature (OCT) compound (VWR, 25608-930) and flash freezing. The tissue blocks were stored at –80 °C before use.

Coverslips were silanized and coated with poly-D-lysine largely following a previous protocol[26]. Briefly, coverslips (Bioptechs, 40-1313-03193) were cleaned with 1:1 37% hydrochloric acid and methanol, silanized in 0.2% (vol/vol) allytrichlorosilane (Sigma, 107778) with 0.1% (vol/vol) triethylamine (Sigma, T0886) in chloroform and stored in dessicated containers for at least 1 d. The coverslips were then coated with 0.1 mg ml^{-1} poly-D-lysine (Santa Cruz, sc-136156) and 1:20,000 orange fluorescent fiducial beads (Invitrogen, F8800) in 1× PBS for 20 min, dried at room temperature for 1 h and then fixed with 4% (vol/vol) PFA (EMS, 15714) at room temperature for 10 min. The coverslips were then washed, dried and stored in the dark at 4 °C in a desiccated container until use.

For cryosectioning, tissue blocks were mounted and equilibrated to –20 °C. Coverslips and Petri dishes were prechilled in the cryostat chamber as well. Slices of 8-µm thickness were cut and melted onto cold coverslips using finger heat and quickly refrozen in the Petri dishes on the bottom of the cryostat chamber. After cryosectioning, the slices were left to dry on the bottom of the cryostat chamber for approximately 2 h. The slices were then fixed in prechilled 4% (vol/vol) PFA in 1× PBS at 4 °C for 20 min followed by two washes with cold 2 mM RVC in 1× PBS and then permeabilized in cold 70% ethanol at 4 °C for at least overnight.

MERFISH sample preparation. The permeabilized ileum samples were prepared for MERFISH imaging largely following a previously published protocol[19]. Briefly, samples were washed in 30% (vol/vol) formamide in 2× SSC twice for 3 min at room temperature then placed on a parafilm-coated Petri dish, and 100 µl of hybridization solution was added on top of the slices. The hybridization solution consisted of 2× SSC, 30% (vol/vol) formamide (Fisher, AM9342), 10% (wt/vol) dextran sulfate (Millipore, S4030), 1 mg ml^{-1} yeast tRNA (Thermo, 15401029), a total concentration of 5 µM encoding probes (see above) and 2 µM of a poly(A) anchor probe (/5Acryd/TTGAGTGGATGGAGTGTAAT T+TT+ TT+T T+TT+ TT+T T+TT+ TT+T T+T, where T+ indicates locked nucleic acid; IDT). To label the boundaries of all cells, a rabbit anti-Na^+/K^+-ATPase primary antibody (Abcam, ab76020) was added to this solution at a final concentration of 3 µg ml^{-1} (1:200 dilution). The sample chamber was then humidified with a piece of moist cotton and sealed with parafilm. Samples were hybridized in a humidified 37 °C oven for 36 to 48 h then washed with 30% (vol/vol) formamide in 2× SSC twice for 3 min at room temperature. Hybridization was then repeated using the same buffers and conditions but with the primary antibody replaced with the goat anti-rabbit oligo-labeled secondary antibody (prepared above) at a final concentration of 4 µg ml^{-1} (1:300 dilution) in a humidified 37 °C oven overnight. Samples were then washed with 30% (vol/vol) formamide in 2× SSC twice for 30 min at 47 °C and covered in 2 mM RVC in Tris-buffered saline (50 mM Tris-HCl, 300 mM NaCl).

After hybridization, samples were embedded in a polyacrylamide gel film, and the protein components were digested away. Briefly, the samples were washed

twice in a gel solution consisting of 4% acrylamide/bis-acrylamide 19:1 (Bio-Rad, 1610144) in 50 mM Tris-HCl, 300 mM NaCl, 0.03% (wt/vol) ammonium persulfate (Sigma, 215589), 0.15% (vol/vol) tetramethylethylenediamine (TEMED; Sigma, T7024) and 2 mM RVC. The coverslips were then inverted on a 65-µl droplet of the gel solution on a Gel Slick (Lonza, 50640)-coated glass plate and allowed to cast for 1 h and 30 min at room temperature. Samples were then gently removed from the glass plate, washed with 2× SSC and digested in a buffer consisting of 2× SSC, 2% (vol/vol) SDS (Thermo, AM9823), 1:100 proteinase K (NEB, P8107S) and 0.25% (vol/vol) Triton-X (Sigma, T8787) at 37 °C for 2 d with a change of digestion buffer after 1 d. Samples were then washed with 2× SSC five times for 30 min at room temperature and imaged immediately or stored at 4 °C.

MERFISH imaging. MERFISH measurements were conducted on a home-built epifluorescence microscope with integrated fluidics control. The microscope was built around a Nikon Ti-2 body and contained a seven-color, solid-state laser illumination source (Lumencor, Celesta) coupled into the microscope with a custom epi-illumination injection system built to our specifications (Lumencor), a pentaband dichroic and a filter cube (Semrock FF421/491/567/659/776-Di01-25x36; FF01-391/477/549/639/741-25; FF01-441/511/593/684/817-25), a motorized *xy* stage (Marzhauser, SCAN IM 130x85), a piezo-controlled objective nano-positioner (Mad City Labs, NanoF2000), a ×60 CFI PlanApo oil objective (Nikon) and two high-performance CMOS cameras (Hamamatsu, Flash 4.0) coupled to the microscope body with a dual-camera splitter (Cairn, TwinCam) and a long-pass dichroic (Semrock, T750lpxrxt-UF2). The focus of the microscope was controlled by a custom autofocus system described previously[19].

MERFISH imaging was performed as described previously[19]. In short, fluorophore readout probes complementary to the readout bit sequences were sequentially hybridized and cleaved off in sequential rounds of imaging, forming the barcode on–off pattern that can be decoded later using a computational pipeline. The readout probes were paired by color and had either Cy5 or Alex647; their sequences have been published previously[19].

The first pair of readout probes and readout probe complementary to the cell boundary immunofluorescence oligo were stained on the bench before imaging. The samples were stained with 3 nM each of the readout probes in 2× SSC, 10% (vol/vol) ethylene carbonate (Sigma, E26258-500G) and 0.25% (vol/vol) Triton-X for 20 min at room temperature followed by 4 µg ml^{-1} DAPI (Fisher, D1306) in the same solution for 10 min at room temperature and washed once with 2× SSC. Coverslips were then mounted in a closed-flow chamber (Bioptechs, FCS2), and buffers were introduced via a home-built fluidics system consisting of four computer-controlled valves (Hamilton, MVP) and a peristaltic pump (Gilson, MP1). The fluidics system controlled the flow of a series of buffers, including a readout hybridization buffer (2× SSC, 10% (vol/vol) ethylene carbonate and 0.25% (vol/vol) Triton-X and 3 nM of the specific readout probes), a readout wash buffer (2× SSC, 10% (vol/vol) ethylene carbonate and 0.25% (vol/vol) Triton-X), a readout cleavage buffer (2× SSC, 50 mM tris(2-carboxyethyl)phosphine) (GoldBio, TCEP25), a cleavage wash buffer (2× SSC) and an imaging buffer (2× SSC, 4 µM Trolox-quinone, 0.5 mg ml^{-1} Trolox, 1:500 rPCO (OYC 46852004) and 5 mM protocatechuic acid (Sigma, 37580-25G-F)).

In each round of imaging, samples were hybridized with readout probes for 15 min, washed with the readout wash buffer for 5 min and then imaged in the imaging buffer. Nine *z* planes were collected at 1.5-µm separation to cover the full volume of the slice. After imaging, the fluorescence signal was extinguished by incubating the sample in the readout cleavage buffer for 15 min. The sample was then washed with the cleavage wash buffer for 5 min and then hybridized for the next round of imaging. Readout probes complementary to the poly(A) anchor probe readout region were imaged in the last round.

Readout probes were imaged with either 635-nm (Cy5) or 750-nm (Alex647) illumination. Orange fluorescent fiducial beads (coated on coverslips) were imaged with 535-nm illumination to align fields of view between imaging rounds. DAPI was imaged with 405-nm illumination, and the poly(A) and cell boundary immunofluorescence signals were imaged with 473-nm illumination.

MERFISH analysis. Individual mRNAs were identified in these data using a previously described analysis pipeline[19]. Briefly, images of the fiducial beads were used to align images of the same field of view from different imaging rounds, and chromatic aberrations between images collected in the 635-nm or 750-nm channels were corrected. Background was removed with a high-pass filter, the image was deconvolved with Lucy–Richardson deconvolution and then noise was removed with a low-pass filter. Individual pixels were assigned to their nearest barcode based on the measured intensity profile across all imaging rounds, assuming that the distance between the nearest barcode and the measured intensity profile falls within a distance smaller than that of a single-bit flip to that barcode. Contiguous pixels assigned to the same barcode were then grouped to form an RNA. Spurious RNAs were removed by filtering over the quality score estimated based on the number of pixels assigned to each RNA as well the average brightness across all rounds for each RNA[19] using the following procedure:

1. All spots were split by the number of pixels n_p assigned to them.
2. For each value of n_p, the spots were ordered by the average brightness b.
3. For each value of n_p and b, the fraction of false-positive measurements was estimated across all spots with the brightness higher than b. This fraction was used as the quality score.
4. All spots with a quality score less than 0.8 were removed.

Single-cell analysis of segmented MERFISH data. Single-cell analysis was performed in scanPy 1.7.2 (ref. [35]). Baysor- or Cellpose[43]-segmented cells were filtered by the following criteria: transcript counts of >10 and <900, and the number of genes expressed is ≥1. Additionally, for Baysor segmentation, the average in-cell confidence is set to >0.75 with a cell area of >50 and <25,000 pixel^3. After filtering, 5,180 of 5,194 cells remained for Baysor segmentation, 7,416 of 8,211 cells remained for Baysor with membrane stain prior and 5,317 of 8,439 cells remained for Cellpose segmentation. All genes were expressed in at least one cell and were kept for subsequent analysis. Gene expression was normalized by cell area; the counts per cell were divided by the cell area and multiplied by a normalization constant (mean(cell area) + 2 × standard deviation(cell area)). Highly variable gene analysis was performed, and 200 of 241 genes were considered highly variable. Fifty PCs were then extracted, and a UMAP embedding was calculated. For clustering, the Leiden algorithm[44] was used, and subclustering was performed on some of the clusters to yield the final sets of clusters.

Reporting Summary. Further information on research design is available in the Nature Research Reporting Summary linked to this article.

Data availability

The following datasets were used in evaluating the developed methods:

1. osmFISH mouse somatosensory cortex[8], 35 genes: http://linnarssonlab.org/osmFISH/availability/.
2. MERFISH mouse preoptic hypothalamus[19], 140 genes: https://doi.org/10.5061/dryad.8t8s248.
3. ISS mouse CA1 region[16], 95 genes: https://doi.org/10.6084/m9.figshare.7150760.v1.
4. STARmap mouse VISp[18], 1,020 genes: https://www.starmapresources.com/data/ (visual_1020, 20180505_BY3_1kgenes).
5. STARmap mouse VISp[18], 160 genes: https://www.starmapresources.com/data/ (visual_160, 20171120_BF4_light).
6. seqFISH$^+$ NIH/3T3 cells[7], 10,000 genes: https://doi.org/10.5281/zenodo.2669683.
7. seqFISH mouse embryo[45], 387 genes: https://marionilab.cruk.cam.ac.uk/SpatialMouseAtlas/.
8. Allen smFISH mouse VISp, 22 genes: https://github.com/spacetx-spacejam/data.
9. MERFISH mouse ileum, 241 genes: https://doi.org/10.5061/dryad.jm63xsjb2.

Code availability

The Baysor package is available at https://github.com/kharchenkolab/Baysor. Baysor parameters for different datasets are reported in Supplementary Table 3. The code to reproduce the results is available at https://github.com/kharchenkolab/BaysorAnalysis/. This repository also contains the links to interactive visualization of the processed datasets using the Vitessce tool (http://vitessce.io/). MERFISH probe design and analysis software is available at https://github.com/ZhuangLab/MERFISH_analysis.

References

29. McInnes, L., Healy, J. & Melville, J. UMAP: uniform manifold approximation and projection for dimension reduction. Preprint at *arXiv* https://arxiv.org/abs/1802.03426v3 (2018).
30. Kanemura, A., Maeda, S. & Ishii, S. Superresolution with compound markov random fields via the variational em algorithm. *Neural Netw.* **22**, 1025–1034 (2009).
31. Blei, D. M., Kucukelbir, A. & McAuliffe, J. D. Variational inference: a review for statisticians. *J. Am. Stat. Assoc.* **112**, 859–877 (2017).
32. Dempster, A. P., Laird, N. M. & Rubin, D. B. Maximum likelihood from incomplete data via the em algorithm. *J. R. Stat. Soc. Series B Stat. Methodol.* **39**, 1–38 (1977).
33. Nielsen, S. F. The stochastic EM algorithm: estimation and asymptotic results. *Bernoulli* **6**, 457–489 (2000).
34. Kimura, T. et al. Expectation–maximization algorithms for inference in Dirichlet processes mixture. *Pattern Anal. Appl.* **16**, 55–67 (2013).
35. Wolf, F. A., Angerer, P. & Theis, F. J. Scanpy: large-scale single-cell gene expression data analysis. *Genome Biol.* **19**, 15 (2018).
36. Zeisel, A. et al. Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq. *Science* **347**, 1138–1142 (2015).
37. Harris, K. D. et al. Classes and continua of hippocampal CA1 inhibitory neurons revealed by single-cell transcriptomics. *PLoS Biol.* **16**, e2006387 (2018).
38. Hodge, R. D. et al. Conserved cell types with divergent features in human versus mouse cortex. *Nature* **573**, 61–68 (2019).
39. Haber, A. L. et al. A single-cell survey of the small intestinal epithelium. *Nature* **551**, 333–339 (2017).

40. Gehart, H. et al. Identification of enteroendocrine regulators by real-time single-cell differentiation mapping. *Cell* **176**, 1158–1173 (2019).
41. Tsoucas, D. et al. Accurate estimation of cell-type composition from gene expression data. *Nat. Commun.* **10**, 2975 (2019).
42. Moffitt, J. R. et al. High-throughput single-cell gene-expression profiling with multiplexed error-robust fluorescence in situ hybridization. *Proc. Natl Acad. Sci. USA* **113**, 11046–11051 (2016).
43. Stringer, C., Wang, T., Michaelos, M. & Pachitariu, M. Cellpose: a generalist algorithm for cellular segmentation. *Nat. Methods* **18**, 100–106 (2021).
44. Traag, V. A., Waltman, L. & Van Eck, N. J. From Louvain to Leiden: guaranteeing well-connected communities. *Sci. Rep.* **9**, 5233 (2019).
45. Lohoff, T. et al. Highly multiplexed spatially resolved gene expression profiling of mouse organogenesis. Preprint at *bioRxiv* https://doi.org/10.1101/2020.11.20.391896 (2020).

Acknowledgements

We thank B. Tasic and B. Long for sharing the non-published Allen smFISH data and aiding in its interpretation and the SpaceTx consortium for facilitating the collaborations. We also thank Y. Boykov (University of Waterloo) for the initial discussions and advice on an alternative segmentation approach based on graph cuts. We are also grateful to a number of colleagues who advised us on the published protocols, including N. Pierson and L. Cai (seqFISH$^+$), S. Codeluppi, L. Borm and S. Linnarsson (osmFISH) and X. Qian, M. Hilscher and M. Nilsson (ISS). Additionally, we thank J. Miller for his input on segmentation benchmarks and B. Lelieveldt for his advising on NCV visualization. We express our gratitude to D. Molchanov and D. Vetrov (HSE, Moscow) for their input on the algorithm. J.R.M. acknowledges pilot funding from the Harvard Digestive Disease Center (P30 DK034854). V.P., P.V.K. and J.R.M. were supported by the Seed Network grant 2019-202743 from the Chan Zuckerberg Initiative. V.P. is funded through a cooperative agreement between University of Copenhagen and Harvard Medical School.

Author contributions

P.V.K. and V.P. formulated the study and the overall approach. V.P. developed the detailed algorithms with advice from R.A.S. and K.K. V.P. implemented the Baysor package. J.R.M., R.J.X. and P.C. developed boundary immunostaining and performed MERFISH measurements. V.P. and P.V.K. drafted the manuscript, with contributions by J.R.M, R.A.S., and R.J.X. All authors provided suggestions and corrections on the manuscript text.

Competing interests

P.V.K. serves on the Scientific Advisory Board to Celsius Therapeutics, Inc., and Biomage, Inc. J.R.M. is a cofounder and Scientific Advisory Board member of Vizgen, Inc. J.R.M. is an inventor on patents associated with MERFISH applied for on his behalf by Harvard University and Boston Children's Hospital. The other authors declare no conflict of interest.

Additional information

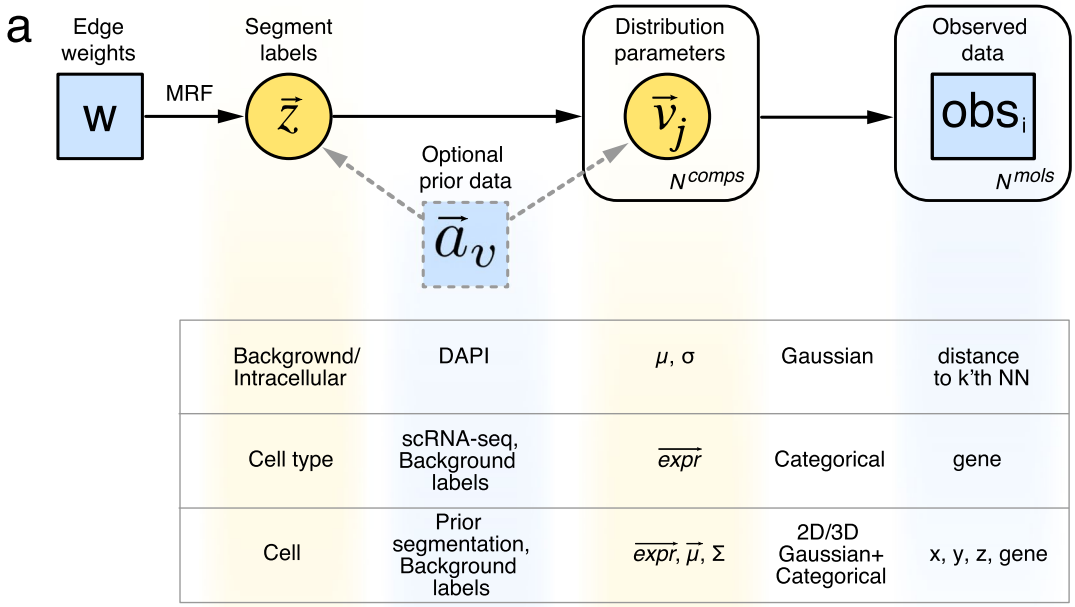
Extended data is available for this paper at https://doi.org/10.1038/s41587-021-01044-w.

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41587-021-01044-w.

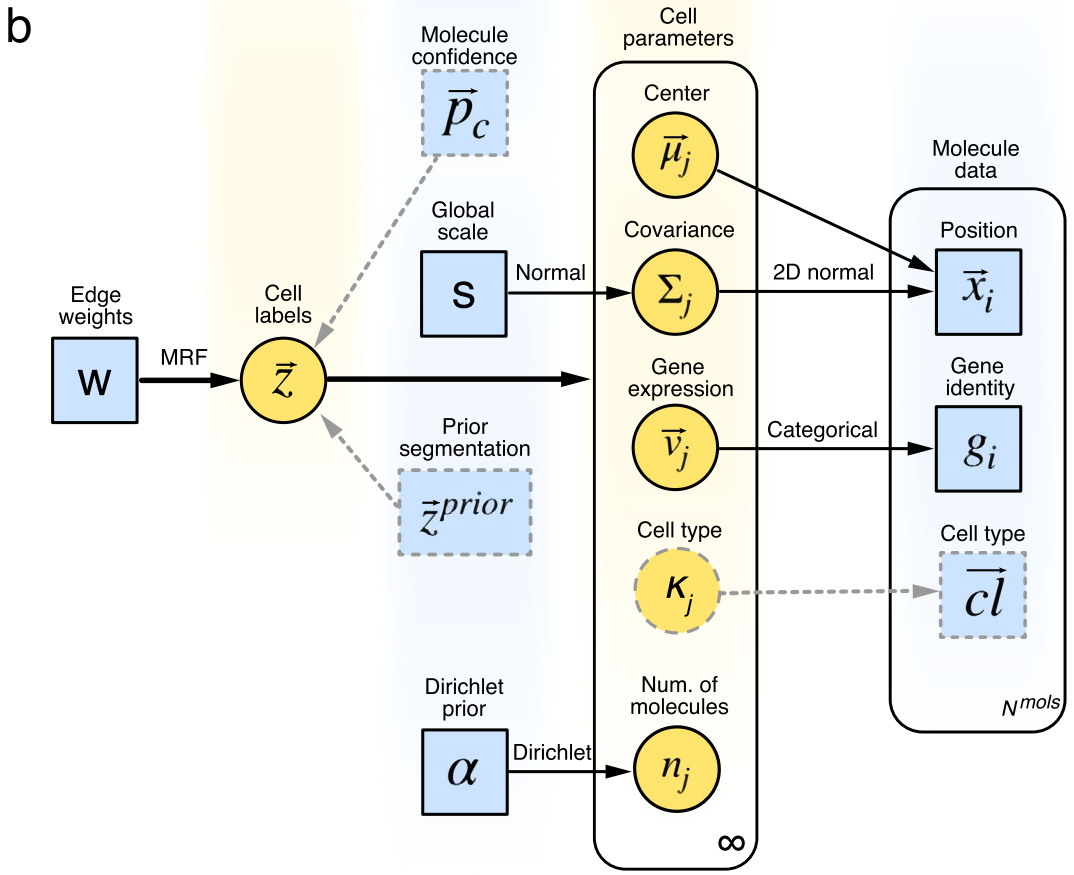
Correspondence and requests for materials should be addressed to Peter V. Kharchenko.

Peer review information *Nature Biotechnology* thanks Kenneth Harris and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

Reprints and permissions information is available at www.nature.com/reprints.

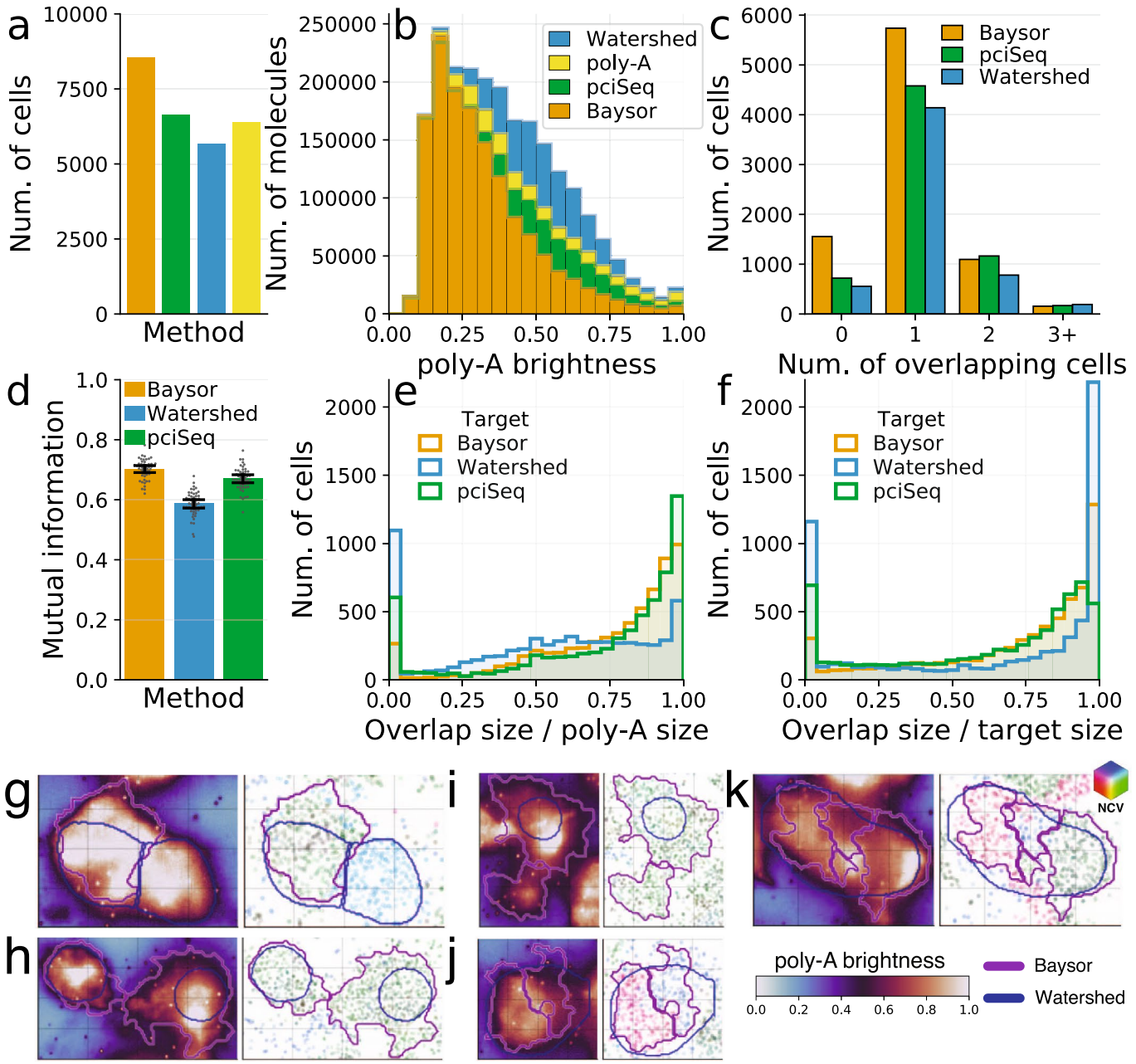


Backgrownd/ Intracellular	DAPI	μ, σ	Gaussian	distance to k'th NN
Cell type	scRNA-seq, Background labels	$\overrightarrow{expr}$	Categorical	gene
Cell	Prior segmentation, Background labels	$\overrightarrow{expr}$, $\vec{\mu}$, Σ	2D/3D Gaussian+ Categorical	x, y, z, gene

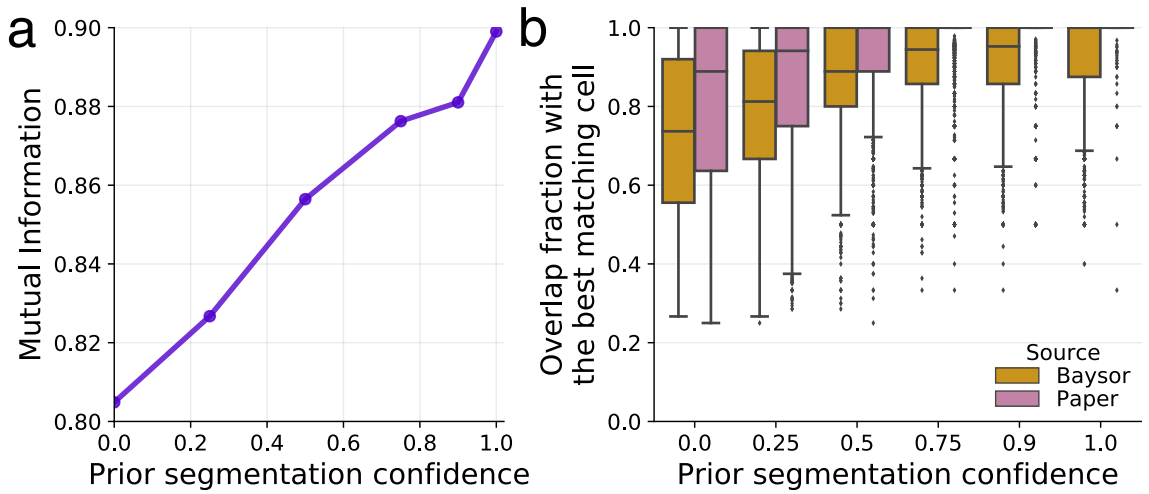


Extended Data Fig. 1 | See next page for caption.

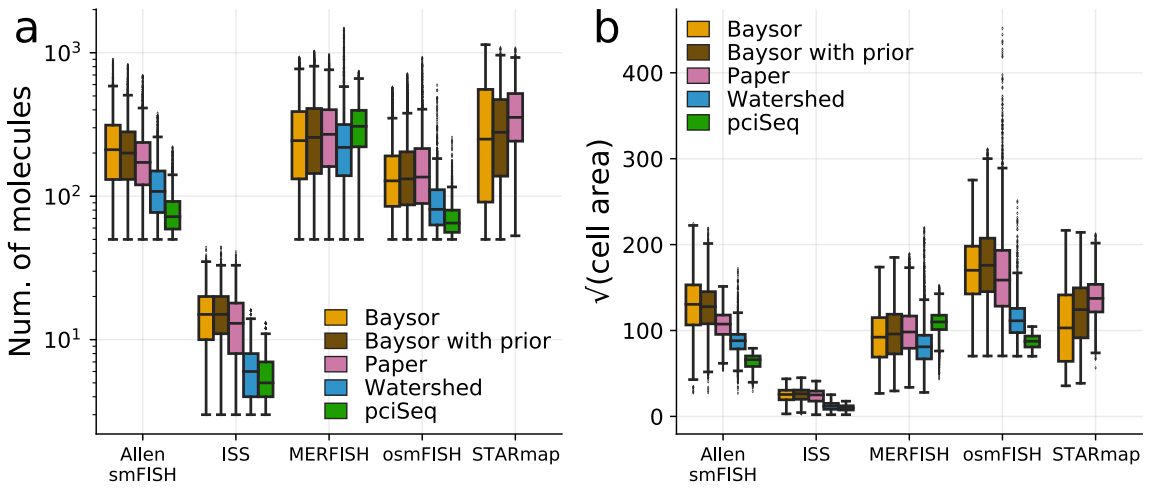
Extended Data Fig. 1 | Graphical models of the segmentation process. Graphical representations of the Bayesian models used for the general Markov-Random Field (MRF) segmentation process (**a**) and the extended model for cell segmentation (**b**) are shown. Blue squares represent input parameters and data for the algorithm. The yellow circles represent the hidden parameters, fitted by the algorithm. Optional input and parameters are shown with dashed border lines. Round-corner boxes represent plate notation for a mixture of distributions with the size of the mixture shown on the bottom right corner. N^{mols} denotes the number of molecules in the dataset, and N^{comps} is the specified number of the mixture components. Arrow labels show the distributions used to model dependencies between the corresponding variables. Matrix variables are shown with the capital letters and vector variables are designated with the overline. **a**, The general MRF model, where the MRF prior with weights **W** is used to account for the spatial dependency of the inferred labels $\overrightarrow{z} \in 1 : N^{comps}$. Examples of the variables and distributions for different labelling problems are noted below the boxes. **b**, The detailed model for the Cell Segmentation problem. Here, Bayesian Mixture Models with Dirichlet prior were used, so the possible number of components of the mixture is infinite, which allows the algorithm to estimate the number of components automatically. To ensure that the components correspond to the actual cells, the Global Scale parameter *s* was introduced, which specifies the expected cell radius.



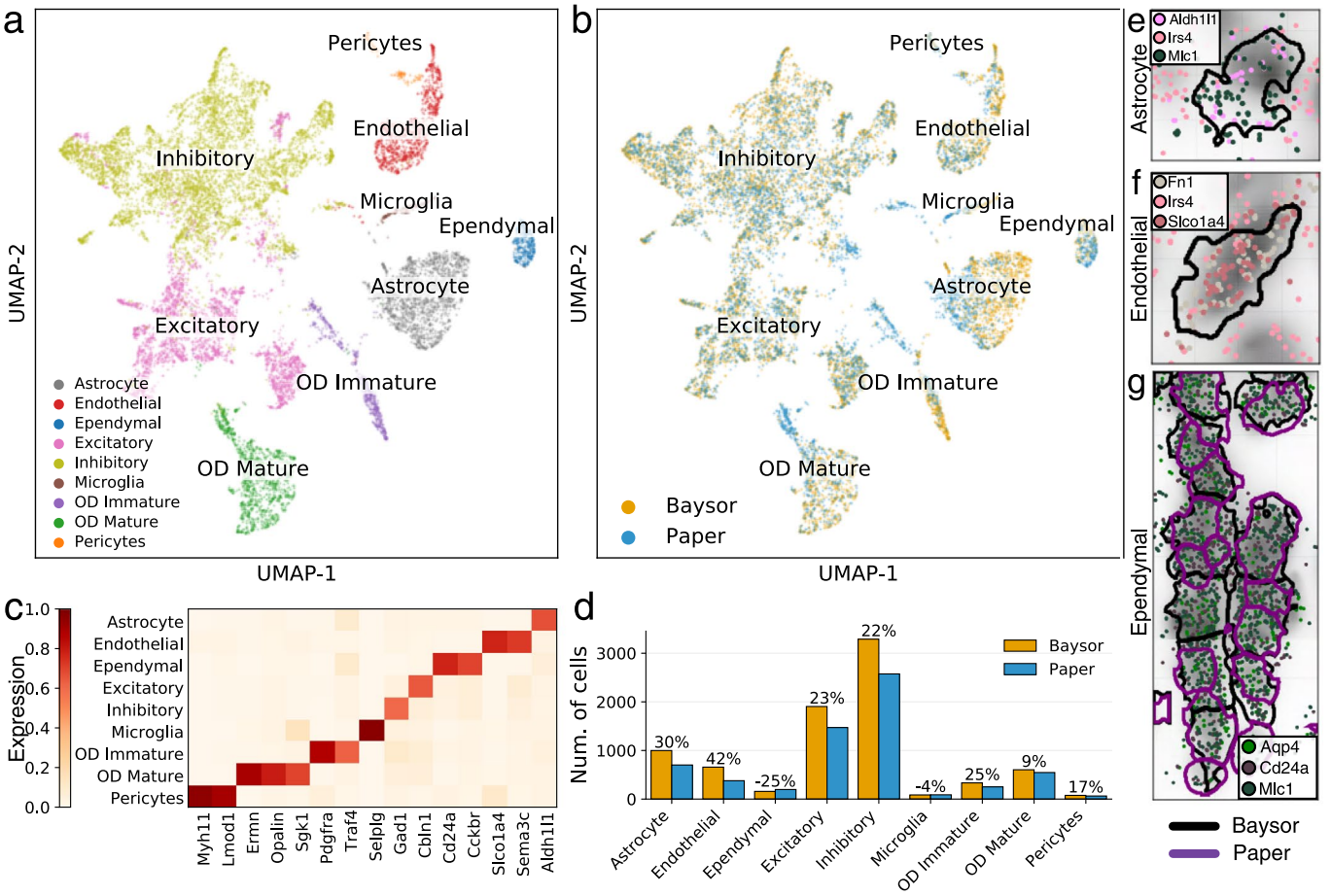
Extended Data Fig. 2 | Comparison of Baysor, pciSeq, and DAPI Watershed segmentations based on poly(A) signal. a, Number of cells in different segmentations. **b**, Distribution of poly(A) brightness (x-axis) across background molecules (that is, molecules outside of the predicted segmentations) for different segmentations (color). Baysor shows the lowest number of transcripts in bright poly(A) regions, while Watershed has the heaviest tail. **c**, Number of cells overlapping with the segmented poly(A) regions is shown as a distribution for different segmentation methods. Baysor shows highest frequency of one-to-one mapping with poly(A) segmentations. **d**, Mutual information (y-axis) of molecule assignment with poly(A) segmentation is shown for different segmentation methods. To account for local variation we split the data over 7x7 grid and showed mean and 95% CI, as well as individual values for $n = 49$ sub-regions. **e**, Size of the overlap for a best-matching cell in the poly(A) segmentation, normalized to the total size (in molecules) of the best-matching poly(A) cell is shown as a distribution for different segmentation methods. Peak near 1.0 indicates that many poly-A cells are fully covered by the best-matching target cells reported by a segmentation method. **f**, Similar to (e), the size of the overlap with best-matching poly(A) cells is shown as a fraction of the target cell size for different target segmentation methods. Peak near 1.0 indicates that many reported target cells are fully covered by the best-matching poly(A) cell. **g-k**, Examples borders of Baysor (purple) and Watershed (blue) segmentations are shown. The left plots show poly(A) signal, while the right plots show molecules colored based on local expression patterns (NCVs). While in most cases there is a good correspondence between the two modalities (**g-i**), in some cases molecular composition clearly indicates presence of distinct cells which are not easily separated from the poly(A) signal intensity (**j,k**).



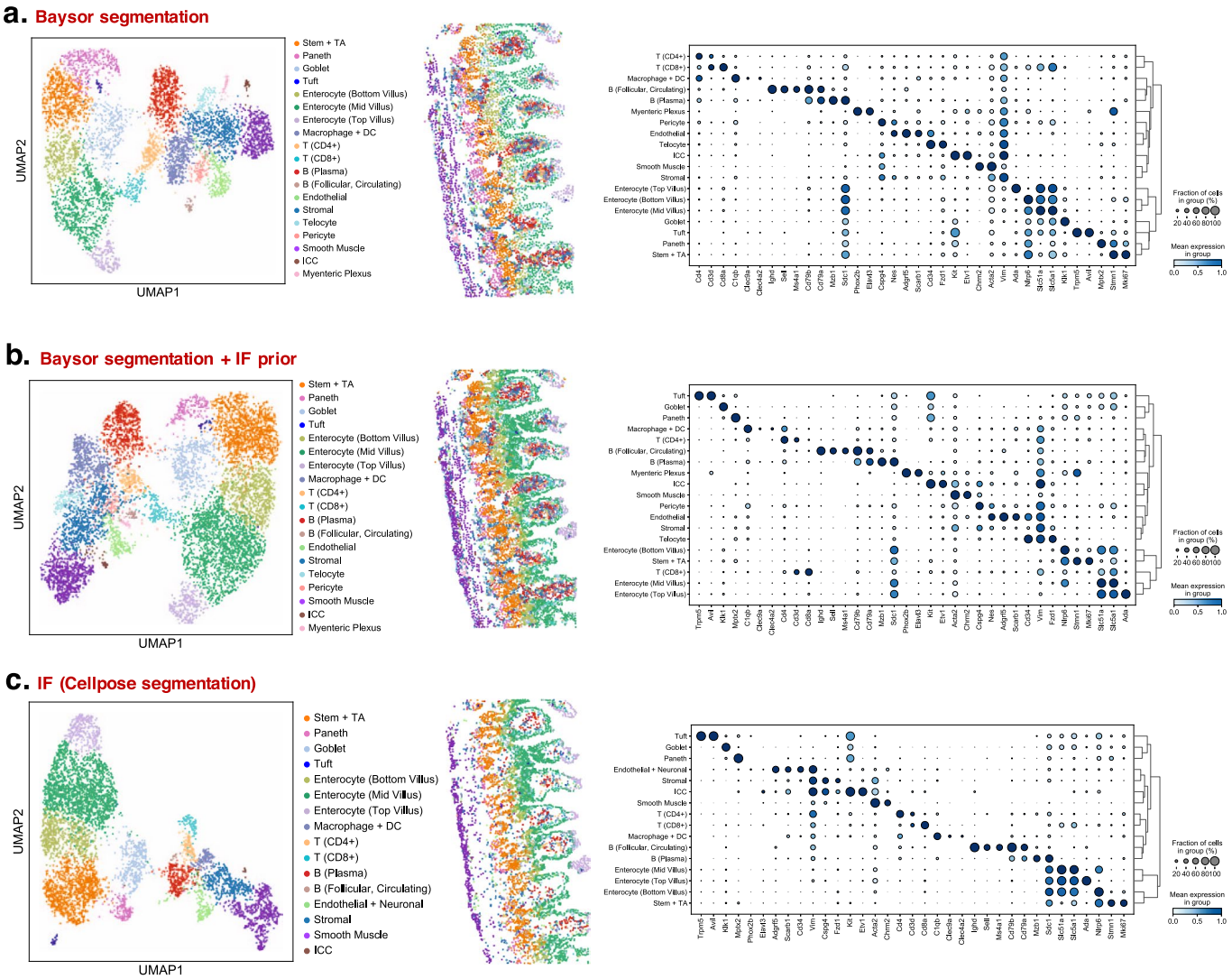
Extended Data Fig. 3 | Impact of the 'prior segmentation confidence' parameter on the difference between the prior and the posterior segmentations on the example of ISS CA1 region. a, The Mutual Information between the Baysor and the Paper (published) segmentations (y-axis) is shown as a function of the prior segmentation confidence (x-axis). Mutual Information does not reach the value of 1.0, as even for prior confidence set to 1.0, Baysor is still allowed to re-assign molecules, recognised as background in the Paper segmentation. **b**, For each cell of the source segmentation (shown with colour), a cell with the largest overlap was picked from the target segmentation. The overlap fraction is shown on the y-axis for the different values of prior segmentation confidence. The boxes represent distribution quartiles with the maximal length of whiskers equal to 1.5 of the inter-quartile range. It can be seen that for high values of the prior confidence, for each Paper cell there is a Baysor cell that covers it completely (confidence $\geq$0.9, Source=Paper). The opposite is not true, as Baysor is allowed to re-assign the background molecules from the Paper segmentation.



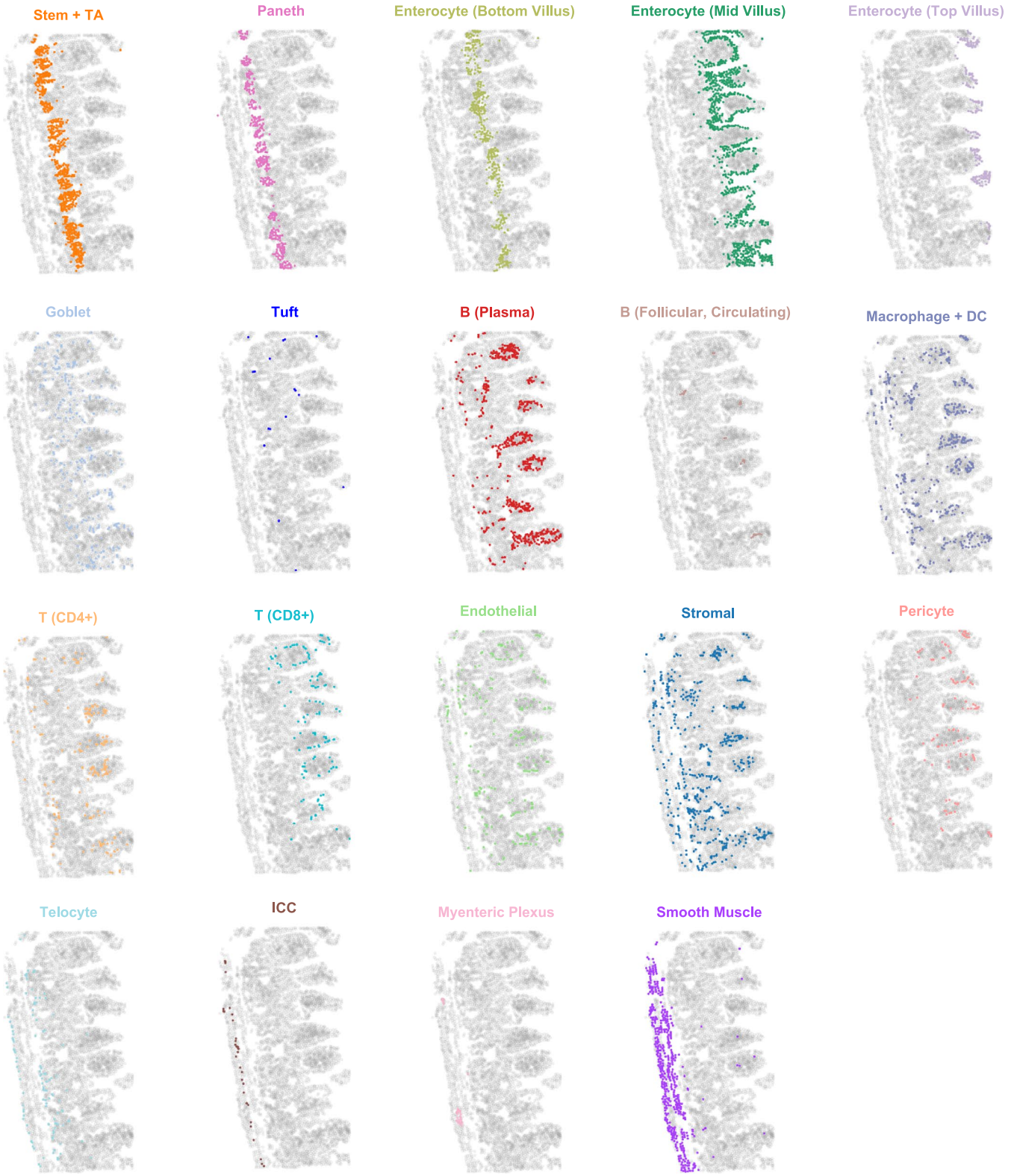
Extended Data Fig. 4 | Cell statistics for different segmentation methods. The boxplots show distributions of the number of molecules per cell (**a**, log-scale y-axis) and the squared root of the cell area, which is an approximation for cell radii (**b**) for different protocols (x-axis) and segmentation methods (fill colours). The boxes represent quartiles with the maximal length of whiskers equal to 1.5 of the inter-quartile range. For all datasets, Baysor has approximately the same values as the published segmentations, which suggests that it is not biased towards over- or under-segmentation. The Watershed and pciSeq methods stably shows lower values, consistent with registering mostly nuclei molecules.



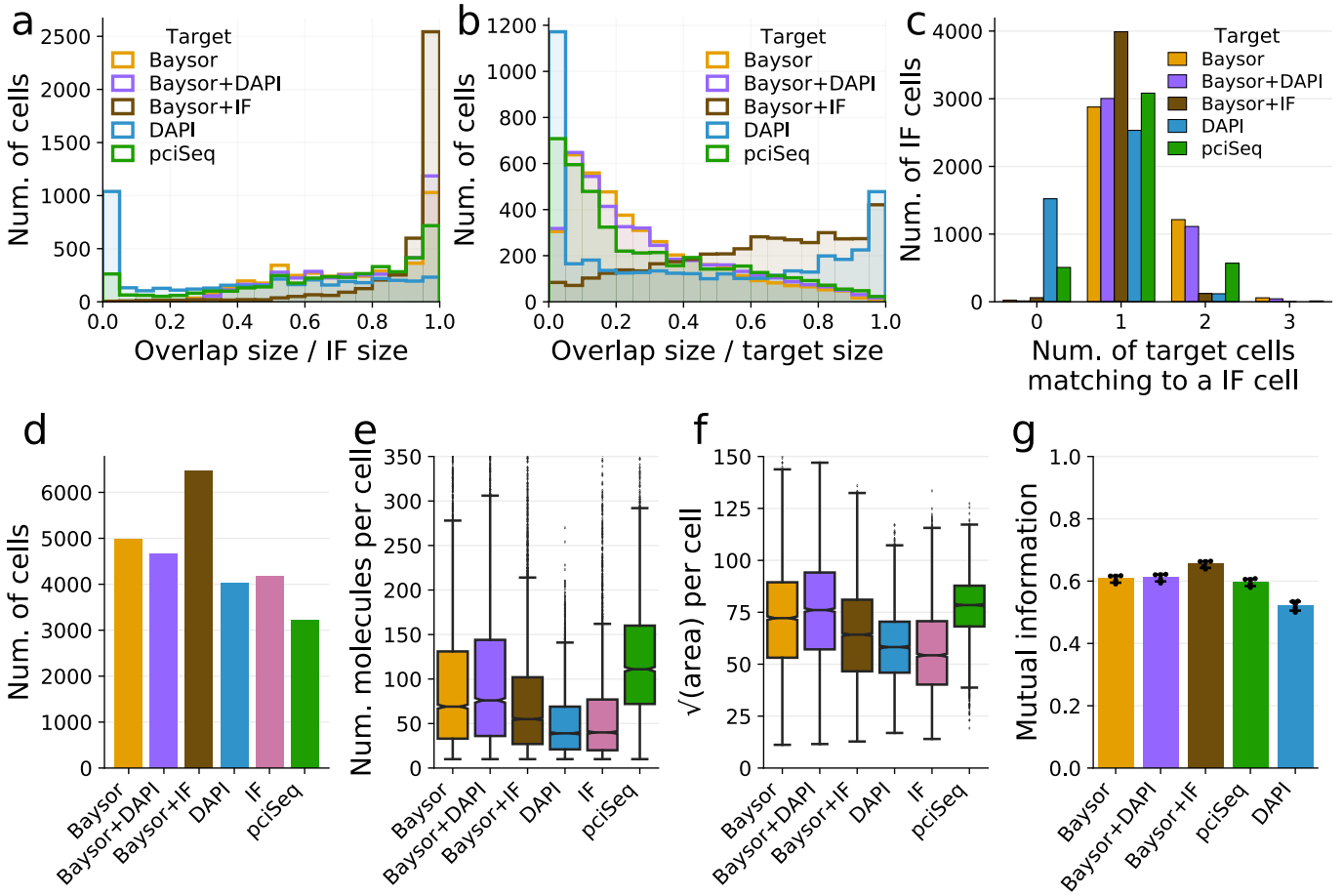
Extended Data Fig. 5 | Comparison of the Baysor and the published segmentation on the MERFISH Hypothalamus dataset. The figure shows the comparison in the same format as Fig. 5. **a**, A joint UMAP embedding of the cells from both Baysor and the paper segmentations. The colors correspond to the annotated cell types. **b**, The same embedding, colored by the segmentation that produced a specific cell. **c**, A heatmap showing expression patterns of marker genes (columns) for the different cell types (rows). The colors show expression levels, normalised for each gene. **d**, The frequency of different cell types is shown for the Baysor (brown bars) and the paper (blue bars) segmentations. The numbers on the top of the bars show excess percentage for Baysor. The largest difference is observed for Endothelial cells, where the Paper segmentation has 42% fewer cells compared to Baysor. **e-f**. Examples of Astrocytes (**e**) and Endothelial (**f**), which were not segmented by the Paper annotation, but were distinguished by Baysor. The dots correspond to the measured molecules, colored by gene (only three the most abundant genes are shown). The grayscale background shows the DAPI signal, and the black contours show the determined cell boundary. **g**. Example of a region with Ependymal cells, showing that for such regions molecules have homogeneous expression patterns. This results in Baysor slightly under-segmenting such cells, which causes the difference in the number of detected cells.



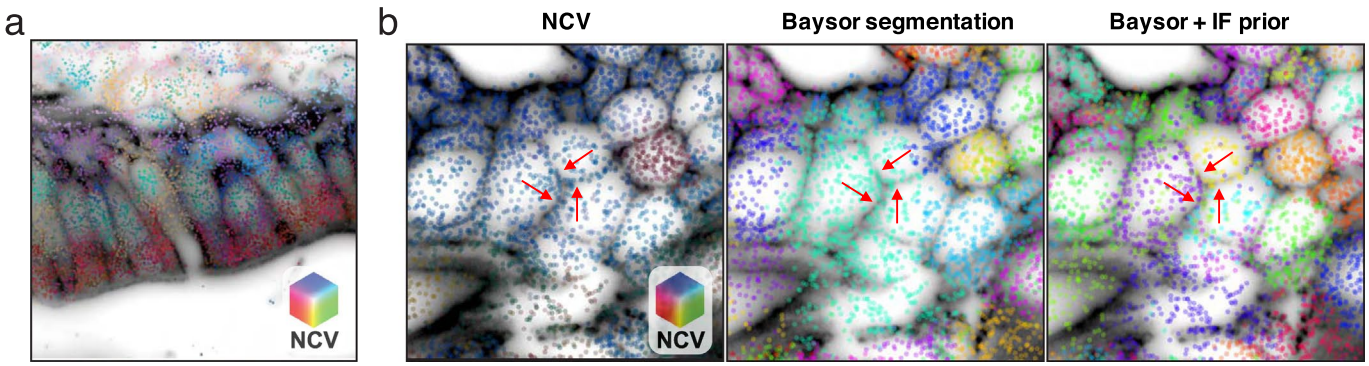
Extended Data Fig. 6 | Comparison of cell clusters in the MERFISH mouse ileum dataset recovered from different segmentation methods. a,b,c, Leiden clusters, cell type spatial distributions, and marker gene expressions in the Na^+K^+-ATPase immunofluorescence (IF) MERFISH mouse ileum dataset, where cells are segmented by (a) Baysor with RNA information only, (b) Baysor with priors provided by Cellpose-derived IF boundaries, (c) Cellpose-derived IF boundaries. Left: UMAP of all identified cells colored based on Leiden clustering. Middle: Spatial distributions of all identified cell clusters colored as in the UMAP. Right: Expressions of marker genes in each of the identified cell clusters. The size of the dots represents the fraction of cells with at least one count of the indicated gene. The color of the dots represents the average expression of each gene across all cell types, log-transformed, and normalized to the cell type with the largest expression. DC: dendritic cells; ICC: interstitial cells of Cajal; TA: transit amplifying cells. **d**, The numbers of each cell type identified by each of the segmentation methods in a-c.



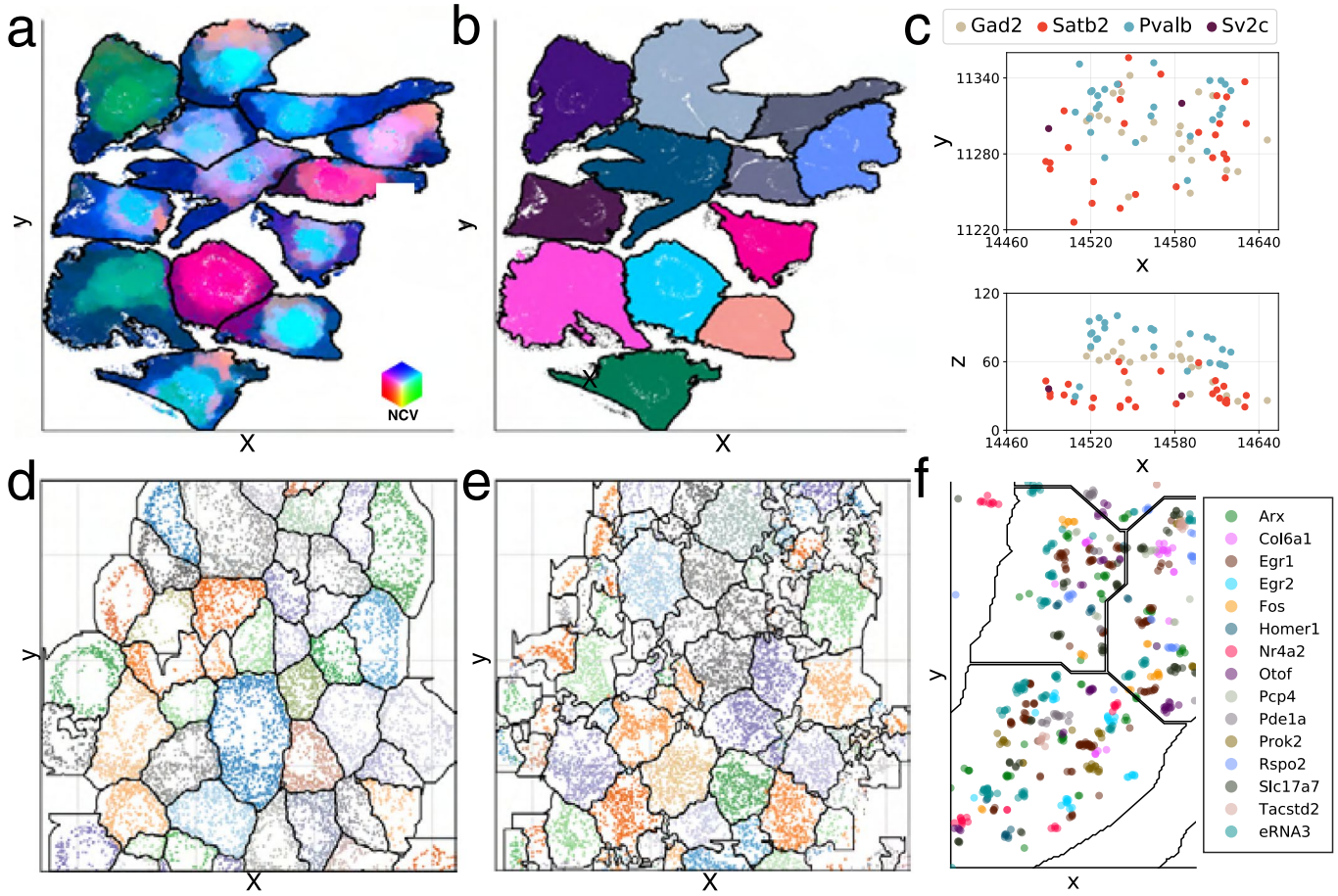
Extended Data Fig. 7 | Spatial distributions of all cell types identified by Baysor (with RNA information only). Gray dots represent the location of all cells, and colored dots represent the location of the indicated cell type. DC: dendritic cells; ICC: interstitial cells of Cajal; TA: transit amplifying cells.



Extended Data Fig. 8 | Additional benchmarks against MERFISH membrane staining data. a, Similar to Fig. 6i of the main manuscript, the distribution shows overlap of different segmentations with membrane IF segmentation. Baysor+DAPI and Baysor+IF correspond to Baysor ran with DAPI and IF segmentations as priors, respectively. **b**, Size of the overlap of different target segmentations with IF segments is shown relative to the size of the predicted cell in the target segmentation. **c**, Distribution of the number of target cells matching to cells of the membrane IF segmentation is shown for different segmentation results. **d**, Number of cells recovered by different segmentation methods **e-f** number of molecules (**e**) and area (**f**) per cell, reported by different segmentation methods. The boxes represent distribution quartiles with the maximal length of whiskers equal to 1.5 of the inter-quartile range. **g**, Agreement between different segmentations and membrane IF segmentation is assessed using mutual information across molecules for $n = 5$ central z-planes. The average and 95% confidence intervals across z-planes, as well as dots for individual values are shown. Only molecules assigned to some cell in any of the methods are used.



Extended Data Fig. 9 | Outstanding challenges: intracellular compartmentalization and homotypic cells. a, An example of intracellular compartmentalization, illustrated by polarized expression pattern of enterocytes in the mouse ileum, as captured by MERFISH. RNAs are colored by NCV. **b**, Example of a homotypic cell cluster from the mouse ileum. Three panels show the same region with membrane IF signal. The left panel shows NCV molecule coloring, whereas center and right panels color molecules assigned to each cell differently. Red arrows point at homotypic cells that Baysor was only able to segment with the help of IF prior.



Extended Data Fig. 10 | Outstanding segmentation challenges. a, Seq-FISH+ Fibroblast[7] data colored by NCVs with black contours showing the published segmentation borders. **b**, The same data, segmented by Baysor with colors showing cell assignment. **c**, Example of cells which are separable only in 3D in the Allen smFISH data. The two plots show 2D projections on the physical x-y and x-z axes correspondingly. Each point represents a molecule, coloured by its gene of origin. Gad2 and Pvalb are markers of inhibitory neurons, while Sv2c with Satb2 are markers of excitatory neurons. These markers are mutually exclusive, and there should be no cell that expresses all four of these markers. **d-e**, Seq-FISH mouse embryo[45] data colored by cell type published cell assignment (**d**) and the Baysor cell segmentation (**e**) with black contours showing the published segmentation borders. It can be seen that the dataset captures cytoplasm-specific genes, lacking nuclei expression, which leads to the holes in the middle of cells. **f**, Example of a cell from the STARmap VISp 160 dataset[18]. The black lines show the published cell boundaries. The plot shows colouring by gene for the 15 most expressed genes.

nature research

Corresponding author(s): Kharchenko

Last updated by author(s): NBT-RA52366B

Reporting Summary

Nature Research wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Research policies, see our Editorial Policies and the Editorial Policy Checklist.

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a	Confirmed	
☐	☒	The exact sample size (*n*) for each experimental group/condition, given as a discrete number and unit of measurement
☐	☒	A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
☒	☐	The statistical test(s) used AND whether they are one- or two-sided *Only common tests should be described solely by name; describe more complex techniques in the Methods section.*
☒	☐	A description of all covariates tested
☒	☐	A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
☐	☒	A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
☒	☐	For null hypothesis testing, the test statistic (e.g. *F*, *t*, *r*) with confidence intervals, effect sizes, degrees of freedom and *P* value noted *Give P values as exact values whenever suitable.*
☐	☒	For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
☒	☐	For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
☐	☒	Estimates of effect sizes (e.g. Cohen's *d*, Pearson's *r*), indicating how they were calculated

Our web collection on statistics for biologists contains articles on many of the points above.

Software and code

Policy information about availability of computer code

Data collection	MERFISH imaging was performed using an open-source microscopy control package available at https://github.com/ZhuangLab/storm-control.
Data analysis	Baysor package is available at https://github.com/kharchenkolab/Baysor. Baysor parameters for different datasets are reported in the Supplementary Table 3. Version 0.5.0 of the package was used. The code to reproduce the results is available at https://github.com/kharchenkolab/BaysorAnalysis/. This repository also contains the links to interactive visualization of the processed datasets using the Vitessce tool (http://vitessce.io/). MERFISH probe design and analysis software is available at https://github.com/ZhuangLab/MERFISH_analysis. Segmentation of imaging data was performed using CellPose v0.6.1 (https://github.com/mouseland/cellpose) and ImageJ v1.53c (https://imagej.nih.gov/ij/download.html).

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors and reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Research guidelines for submitting code & software for further information.

Data

Policy information about availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A list of figures that have associated raw data
- A description of any restrictions on data availability

The following datasets were used in evaluating the developed methods:
osmFISH mouse somatosensory cortex, 35 genes: http://linnarssonlab.org/osmFISH/availability/
MERFISH mouse preoptic hypothalamus, 140 genes: https://doi.org/10.5061/dryad.8t8s248

ISS mouse CA1 region, 95 genes: https://doi.org/10.6084/m9.figshare.7150760.v1
STARmap mouse VISp, 1020 genes: https://www.starmapresources.com/data/
STARmap mouse VISp, 160 genes: https://www.starmapresources.com/data/
seqFISH+ NIH/3T3 cells, 10000 genes: https://doi.org/10.5281/zenodo.2669683
seqFISH mouse embryo, 387 genes: https://marionilab.cruk.cam.ac.uk/SpatialMouseAtlas/
Allen smFISH mouse VISp, 22 genes: https://github.com/spacetx-spacejam/data
MERFISH mouse ileum, 241 genes: https://doi.org/10.5061/dryad.jm63xsjb2

Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☒ Life sciences ☐ Behavioural & social sciences ☐ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/documents/nr-reporting-summary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size	One slice of the mouse ileum was sufficient to illustrate the ability of Baysor to identify cells in this cell-dense tissue.
Data exclusions	Segmented cells that were outliers in size (<50 pixelA3 or> 25,000 pixelA3) or cells in which the majority of assigned RNAs were assigned with low confidence were excluded from downstream clustering. Cells that fell outside of these cuts were representative of cell segmentation errors.
Replication	MERFISH measurements in the mouse ileum were replicated six times (two replicates from three animals) and all produced RNA abundances that correlated strongly with those for the data presented here.
Randomization	Randomization was not relevant for our study as the mouse ileum data were used only to validate the ability of Baysor to segment cells in this cell-dense tissue rather than to report on any correlate with mouse phenotype, genotype, or treatment.
Blinding	Blinding was not relevant for our study as we did not treat the mice in a way designed to induce a measurable effect.

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

Materials & experimental systems

n/a	Involved in the study
☐	☒ Antibodies
☒	☐ Eukaryotic cell lines
☒	☐ Palaeontology and archaeology
☐	☒ Animals and other organisms
☒	☐ Human research participants
☒	☐ Clinical data
☒	☐ Dual use research of concern

Methods

n/a	Involved in the study
☒	☐ ChIP-seq
☒	☐ Flow cytometry
☒	☐ MRI-based neuroimaging

Antibodies

Antibodies used	1: Goat-anti-rabbit IgG (Thermo A16112, Lot 70-54-072919). 2. Rabbit anti-Na/K-ATPase (Abeam ab 76020, Lot GR3237646-18). 3. The oligo-labeled Goat-anti-rabbit IgG was produced as described in methods from the antibody listed above.
Validation	The primary antibody has been validated for immunofluorescence in mouse by the manufacturer. We further validated this antibody and the secondary antibody by showing that our immunofluorescence staining produces the expected cell-surface staining in mouse gut slices.

Animals and other organisms

Policy information about studies involving animals; ARRIVE guidelines recommended for reporting animal research

Laboratory animals	C57BL/6, male mice, specific pathogen free, aged 8-12 weeks were housed under regular dark/light cycles at ambient temperature and humidity.
Wild animals	No wild animals were used.
Field-collected samples	No field-collected samples were used
Ethics oversight	The Harvard Medical School Office of the IACUC approved animal protocols.

Note that full information on the approval of the study protocol must also be provided in the manuscript.