Published in final edited form as:
Nat Methods. 2020 January ; 17(1): 101–106. doi:10.1038/s41592-019-0631-4.

Probabilistic cell typing enables fine mapping of closely related cell types *in situ*

Xiaoyan Qian[#1], **Kenneth D. Harris**[#*,2], **Thomas Hauling**[1,2], **Dimitris Nicoloutsopoulos**[2], **Ana B. Muñoz-Manchado**[3], **Nathan Skene**[2,3], **Jens Hjerling-Leffler**[3], **Mats Nilsson**[*,1]

[1]Science for Life Laboratory, Department of Biochemistry and Biophysics, Stockholm University, Stockholm, Sweden

[2]Institute of Neurology, University College London, London, UK

[3]Department of Medical Biochemistry and Biophysics, Karolinska Institutet, Stockholm, Sweden

[#] These authors contributed equally to this work.

Abstract

Understanding the function of a tissue requires knowing the spatial organization of its constituent cell types. In the cerebral cortex, single-cell RNA sequencing (scRNA-seq) has revealed the genome-wide expression patterns that define its many, closely related neuronal types, but cannot reveal their spatial arrangement. Here we introduce *probabilistic cell typing by in situ sequencing* (pciSeq), an approach that leverages prior scRNA-seq classification to identify cell types using multiplexed *in situ* RNA detection. We applied this method by mapping the inhibitory neurons of hippocampal area CA1, for which ground truth is available from extensive prior work identifying their laminar organization. Our method identified these closely related classes in a spatial arrangement matching ground truth, and further identified multiple classes of isocortical pyramidal

[*]Correspondence: kenneth.harris@ucl.ac.uk; mats.nilsson@scilifelab.se.

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

Data availability
Analysis files are available at https://doi.org/10.6084/m9.figshare.7150760.v1, and an interactive online viewer is at http://insitu.cortexlab.net. The raw image files are available from corresponding authors upon reasonable request.

Code availability
Code for ProMMT algorithm in gene selection is available at https://github.com/cortex-lab/Transcriptomics. Code for probe design is available at https://github.com/Moldia/multi_padlock_design. MATLAB Code for image analysis and cell typing is available at https://github.com/kdharris101/iss. A Python version of the cell-calling algorithm, designed to work with StarFISH data standards, is available at https://github.com/acycliq/cell_call. All custom code is freely accessible.

Author contributions
XQ wrote DNA probe design software, performed experiments, analyzed data, designed *in situ* sequencing protocol, prepared figures, wrote manuscript. KDH conceived the study, designed and wrote analysis software, wrote manuscript. TH designed *in situ* sequencing protocol. DN designed and wrote online web viewer, performed simulations, and wrote Python translation of cell calling code. ABMM designed tissue preparation protocols and provided samples. NS contributed to gene panel selection. JHL conceived the study and supervised tissue sample preparation and collection. MN conceived the study, designed *in situ* sequencing protocol, supervised experiments, wrote manuscript.

Competing interests
XQ, TH, MN hold shares in Cartana AB, a company that commercializes *in situ* sequencing reagents.

cell in a pattern matching their known organization. This method will allow identifying the spatial organization of fine cell types across the brain and other tissues.

Introduction

Bodily tissues are composed of a myriad variety of cell types, which differ in their spatial organization, morphology, physiology, and gene expression. Different varieties of cells can be distinguished by differences in their transcriptomes, and spatially resolved transcriptomic methods raise the possibility of mapping cellular varieties at large scale [1]. While transcriptional differences between some varieties are clear cut, others can be subtle. In the cerebral cortex, the genes expressed by neurons differ greatly from those expressed by multiple classes of glia [2–8], but there exists a remarkable diversity of finely-related neuronal subtypes, particularly among inhibitory interneurons, whose transcriptomes may differ by only a few genes. Thus, while the diversity of cortical cells was known to classical neuroanatomists, accurately relating fine transcriptomic varieties to classically defined cortical neurons has proved challenging.

To validate that spatial transcriptomic analyses can genuinely distinguish finely-related cell types, it is essential to work in a system where ground truth is available from prior work with other methods [9–11]. The interneurons of hippocampal area CA1 provide a unique such opportunity: several decades of work using methods of anatomy, immunohistochemistry and electrophysiology have identified around 20 interneuron subtypes, which are arranged in a stereotyped spatial organization, differ in their computational function, and expression of marker genes [12–14]. Analysis of CA1 interneuron classes by single-cell RNA-sequencing (scRNA-seq) yields clusters strikingly consistent with these classically-defined types [6]. Mapping the spatial organization of CA1 interneurons is thus not only important to understand the brain’s memory circuits, but also provides a powerful way to validate spatial cell type mapping approaches for closely related subtypes, using the spatio-molecular ground truth provided by this system.

Here we provide a spatial map of CA1 interneuron types, using a new approach to *in situ* cell typing based on *in situ* RNA expression profiling. While several approaches to multiplexed *in situ* RNA detection and cell type classification have been proposed [9,15–17], none have yet shown the ability to distinguish fine cortical cell types known from prior ground truth. Here we introduce *probabilistic cell typing by in situ sequencing* (pciSeq), a method with several advantages over other methods. Because it uses low-magnification (20x) imaging, it enables large regions to be analyzed quickly and with reasonable data sizes. Because our chemical methods have very low misdetection rates, our analysis methods can confidently identify cell classes from just a few detections of characteristic RNAs. Finally, because our cell calling algorithms yield probabilistic readouts, they are able to report the depth to which it is able to confidently classify cells. We show that this combination allows cell typing of closely-related neuronal classes, verified by the ground truth available from CA1’s laminar architecture.

Results

CA1 interneurons constitute around 20% of CA1 neurons and thus around 5% of CA1 cells. To rigorously test pciSeq, we focused on distinguishing fine subtypes within this 5% rather than the easier problem of finding major differences within the remaining 95%.

The pciSeq method consists of three steps (Supplementary Figure S1). First, we select marker genes sufficient for identifying cell types, using previous scRNA-seq data. Second, we apply *in situ* sequencing to detect expression of these genes at cellular resolution in tissue sections. Third, gene reads are assigned to cells, and cells to types using a probabilistic model derived from scRNA-seq clusters.

Gene panel selection

To select a gene panel, we developed an algorithm that searches for a subset of genes that can together identify scRNA-seq cells to their original clusters, after downsampling expression levels to match the lower efficiency of *in situ* data (see Online Methods). The gene panel was selected using a database of interneurons from mouse hippocampus [6] (Supplementary Figure S2) as well as isocortex [3], and the results were manually curated prior to final gene selection, excluding genes likely to be strongly expressed in all cell types even if at different levels, and favoring genes which have been used in classical immunohistochemistry (Supplementary Table S1, Supplementary Figure S3). Although our focus was on interneurons, we included some genes identifying CA1 excitatory cells (e.g. *Wfs1)* as well as oligodendrocytes (*Plp1).* A further set of three genes were excluded after initial experiments, as their expression was widespread in neuropil and did not help identify cell types (*Slc1a2*, *Vim*, *Map2*). The final panel contained 99 genes.

In situ sequencing

To generate RNA expression profiles, we modified the *in situ* sequencing method described by Ke *et al.* [18] (Supplementary Figure S4, Supplementary Methods). Padlock probes were designed for the selected genes, each containing two arms together matching a 40-basepair sequence on the cDNA; a 4-basepair barcode; an “anchor sequence” allowing all amplicons to be labelled simultaneously; and a 20-basepair hybridization sequence for additional readouts. For weakly expressed genes, we designed probes matching multiple target sequences along the mRNA length, which aided their detection without compromising detection of others (Supplementary Figure S5). In total we designed 755 probes for 99 genes, but used only 161 barcodes out of 1024 (=4^5) possible combinations to allow error correction (for probe sequence and barcodes see Supplementary Table S2).

To apply the method *in situ*, mRNA is enzymatically converted to cDNA and then degraded. The padlock probe library is applied, and a ligase circularizes probes which are then rolling-circle amplified, generating sub-micron sized DNA molecules (rolling-circle products: RCPs), each carrying hundreds of copies of the probe’s barcode. The barcodes are identified with an epifluorescence microscope with 20x objective in five rounds of multi-color imaging (Figure 1A). Finally, RCPs for two genes which express strongly (*Sst* and *Npy*) are detected separately in a 6th round by hybridizing fluorescent probes to their target recognition sequences. Data are analyzed using a custom pipeline, including point-cloud registration to deal with chromatic aberration in the images, and compensation for optical or chemical crosstalk between bases in the sequencing readout (Figure 1B; Supplementary Figure S6, F and G and Online Methods). These improved chemical and analytic methods achieved a density of reads sufficient for fine cell type assignment.

Our first experiments were performed targeting a subset of 84 genes on four coronal sections of mouse brain (10 μm fresh frozen). After verifying that detected expression patterns match *in situ* hybridization data from the Allen Mouse Brain Atlas [19], we continued with two further experiments using the full 99-gene panel, on two and eight coronal sections, respectively. All 14 sections were from one P25 male CD1 mouse and covered different parts of the dorsal hippocampus (Supplementary Figure S7). Each section contained roughly 120,000 cells and in total 15,424,317 reads passed quality control (Supplementary Table S3). We displayed each read with symbols whose colors grouped genes often expressed by similar cell types, and glyph distinguished genes within these color groups (Figure 1, C and D).

Expression patterns were consistent with expectation at multiple levels of detail. Expression differed between regions (Figure 1C, top), for example with the inhibitory thalamic reticular nucleus dominated by inhibitory-associated genes (blue) and the CA1 pyramidal layer dominated by pyramidal-associated genes (green). Zooming in to the hippocampus (Figure 1C, 2nd row) revealed differences between cell layers and zooming further to single neurons (bottom two rows) showed genes grouped together in combinations expected from scRNA-seq. Expression patterns of genes present in the Allen Mouse Brain Atlas [19] matched at a corresponding coronal level (examples in Figure 1E). Read densities were consistent between experiments, even with different gene panels, further supporting the reliability of the technique (r = 0.93; Supplementary Figure S8A). We manually drew hippocampal CA1 regions (Supplementary Figure S9) and used pciSeq approach to identify the cell types of 27,338 CA1 neurons, from 28 hippocampus sections. Data files for all experiments are available at https://doi.org/10.6084/m9.figshare.7150760.v1, and an online viewer showing reads and probabilistic cell type assignments is at http://insitu.cortexlab.net.

Probabilistic cell typing

A fundamental challenge for *in situ* cell typing is assigning genes to cells, as boundaries between cells are difficult to obtain in 2D imaging. We counterstained all sections with DAPI to reveal nuclei; standard watershed segmentation yielded boundaries containing many, but not all the genes belonging to them (Figure 2A). To solve this problem, we developed a Bayesian algorithm which leverages scRNA-seq data to simultaneously estimate the probability of assigning each read to each cell, and each cell to each class. (Figure 2A, straight lines; Supplementary Figure S10). Note that the algorithm does not take into account a cell’s laminar location, allowing this to be used later for independent validation.

The algorithm mapped CA1 cells to 70 fine classes (previously defined by scRNA-seq clustering, and including pyramidal cells and some non-neurons), however laminar ground truth from previous work is usually only available for a coarser level of classification. Therefore, validating the results of pciSeq against anatomical ground truth data required that the fine cell classes be merged into coarser "superclasses" (Supplementary Table 4). These include 16 interneuron classes: 3 types of interneuron-selective cell; 2 types of *Cck* cell; 2 types of neurogliaform (NGF) cell; 2 types of GABAergic projection cell; 3 types of parvalbumin cell and 4 types of somatostatin cell (Supplementary Tables S4 and S5).

To represent the results on a spatial map, we displayed each cell's class assignments by a pie-chart, of size proportional to total gene count, with the angle of each slice indicating the probability of assignment to a fine transcriptomic class, and slices color-coded according to their superclass assignments (Figure 2; see also Supplementary Figure S11; for all 28 cell type maps, see Supplementary Results; online viewer at http://insitu.cortexlab.net). Although our panel was aimed at distinguishing interneurons, we also obtained confident distinction of two types of pyramidal cell inside and outside of CA1. Non-neuronal cells however could not be distinguished from each other, as our panel did not contain genes to separate them; indeed, many non-neurons had no gene reads at all, and were therefore assigned as unclassified. The average number of gene reads per cell was over 20 for most targeted cell types, and the number of unique genes detected per cell was in the range 5 to 10 (Figure 3A). The probabilistic algorithm allows diagnostics showing which genes provided evidence for calling as one type over another (Supplementary Figure S12).

Validation of cell typing

The algorithm's cell type assignments conformed closely to known combinatorial patterns of gene expression in CA1 interneuron subtypes. Across all experiments, the patterns of both classical and novel interneuron markers were consistent with scRNA-seq results, as well as the known biology of CA1 interneurons (Supplementary Discussion; Supplementary Figure S13). Moreover, the cell type composition was consistent between the left and right hemispheres (Supplementary Figure S8B).

We validated pciSeq, as well as the scRNA-seq classification it relies on, by verifying that cell classes it identifies are found in appropriate layers. The layers in which cell types were identified were consistent with known ground truth (Supplementary Discussion; Figure 3C). This close correspondence with independent studies verifies that the method can accurately identify biological cell types, across a wide dynamic range of cell abundances, ranging from very rare subtypes (IS2 and *Sst/Nos1*, Supplementary Figure S14) to types with thousands per section (PC CA1) (Supplementary Table S5, Supplementary Figure S8).

As a further validation of the cell calling, we performed an analysis of error rates in simulated data. To do so, we replaced the actual read distributions with simulations subsampled from cells in the scRNA-seq database, for which cell type information is therefore available down to the finest details (Supplementary Methods). This analysis showed that with the current detection efficiency and false positive rate, cells could be reliably assigned to fine inhibitory classes comprising as little as ~0.5% of all cells in the tissue (Supplementary Figure S15).

To evaluate the minimal number of genes needed for the pciSeq algorithm to correctly classify cells, we also compared the relative accuracy of cell classification at different gene panel sizes (Supplementary Figure S16). The analysis showed the importance of having relevant genes rather than having high numbers of genes. When genes were added in optimal order, coarse cell types were classified from the top 50 genes similarly to how they were classified by the full panel; for identification of fine cell types, around 70 genes were needed. When genes were added in a random order, however, performance increased more slowly, reaching equivalent performance only when the whole panel was included. Thus, accurate classification of fine cell types can be obtained with modest-size gene panels, but only if they are chosen carefully.

Application of the method in the isocortex

To verify that the method can also work in structures for which it was not directly optimized, we applied the same method to map neurons of the isocortex. Although not specifically designed to distinguish isocortical excitatory and inhibitory cell types, the panel nevertheless contained several genes that distinguish them.

We took cell type definitions from the scRNA-seq data published by Zeisel *at al.*[8], using all neuronal types that the authors annotated to be present in those cortical regions found in the coronal section analyzed (isocortex; cingulate/retrosplenial; and piriform). We mapped 11 000 cells distributed across 15 excitatory and 10 inhibitory classes (Supplementary Figure S17). As in CA1, the frequencies of different neuronal types ranged from a handful for the rare ones, to thousands for the most frequent, and was similar in the two hemispheres (Supplementary Figure S17B). Although ground-truth information on the laminar organization of inhibitory classes is not available as it is in CA1, we were able to recapitulate the laminar organization of excitatory cells in isocortex, as well as between distinct cortical regions in the section (Supplementary Figure S17, C and E).

Discussion

We have presented pciSeq, a method for probabilistic cell typing based on *in situ* sequencing data. We validated the method by mapping interneurons in hippocampal area CA1, a group of closely related neuronal types that together comprise approximately 5% of the cells in this region. We found that the method was able to confidently classify fine subtypes representing as little as 0.5% of the total cells in the region. Furthermore, assigning these fine transcriptomic classes to 18 biological superclasses for which laminar ground truth was available, we confirmed that the spatial assignments made by pciSeq were accurate.

There exist multiple methods for multiplexed *in situ* RNA detection and cell calling [9,15–17,20], each of which presents various advantages and disadvantages. At a computational level, our method's key advantages are its probabilistic assignment of cells to classes, which indicates the confidence and depth with which the cells can be classified, and its probabilistic assignment of reads to cells, avoiding problems of uncertain segmentation. At the chemical level, our method's key advantage is its low false-positive gene detection rate. This low false-positive rate means that even one or two reads of an RNA can provide strong evidence for a cell to belong to a particular class. Thus, while the method has higher false-negative rates than FISH-based approaches, classification of cell types can still confidently be performed by designing a panel of genes that are expressed strongly enough to ensure enough reads of each. The lower read density of the current method provides a complementary advantage over FISH-based methods: it uses 20x objective for faster imaging and reduction in data size compared to 60x-100x imaging for single-molecule FISH [16,17,21], and allowing entire mouse brain sections to be processed.

The pciSeq method requires that scRNA-seq data be available for the cell system of interest, and that cluster analysis has been run on this data. These scRNA-seq clusters are used to design the gene panel, and the algorithm's output is a probabilistic assignment of each *in situ* cell to these scRNA-seq clusters. Although our primary test of the method was to a very well understood cell system with laminar ground truth, this is not necessary to apply the method, only to validate it: pciSeq does not require the scRNA-seq varieties to have been identified with known cell types. Indeed, using the same gene panel that we selected from a clustering of CA1 inhibitory neurons, pciSeq was able to correctly map isocortical and piriform excitatory cells to clusters taken from an independent whole-nervous-system dataset [8]. Thus, the method should be applicable to any tissue where scRNA-seq data is available. Large-scale scRNA-seq projects are now underway for the whole body, and the data required to design panels and apply this method to all tissues will soon be available. The pciSeq approach requires only low-magnification imaging, and so may be applied high throughput, raising the possibility of body-wide spatial cell type maps in the near future.

Online Methods

Gene selection

We chose the gene panel for *in situ* sequencing using an automated algorithm based on scRNA-seq data. The algorithm was run on data from CA1 [2,6] and isocortex [3], restricting in both cases to GABAergic neurons, our cell type of primary interest. The final panel was selected by manual merging and curation of the automatically generated lists. During this manual stage, we excluded genes that were expressed in all classes (even if at different mean levels), and also added some genes used in classical immunohistochemical analysis of CA1 inhibitory cells. These latter genes were not essential for accurate cell typing: the algorithm performed comparably well when they were excluded from analysis (Supplementary Figure S18), and furthermore the same gene accurately identified isocortical pyramidal cells (Supplementary Figure S17), for which no genes were manually selected.

The algorithm starts by clustering the scRNAseq data, for which we used a probabilistic algorithm called ProMMT [6]. Other clustering algorithms could be used also, however for optimal functioning of the pciSeq cell typing algorithm it is recommended to use algorithms for which within-cluster distributions of gene expression are not strongly bimodal, so can be reasonably modeled by a negative binomial distribution. This results in a cluster assignment k_c for each cell c, from which one can compute the mean expression $\mu_{g,k}$ for each gene g and cluster k. We then clustered mean vectors $\boldsymbol{\mu}_k$ hierarchically, yielding a representation of each cluster k as a leaf of a binary tree.

To automatically select genes for *in situ* analysis, we used a combinatorial search algorithm, that optimized a score function over possible gene sets $\mathbb{G}$. Given a set of genes $\mathbb{G}$, we reassigned each cell c to a cluster $k'_{c;\,\mathbb{G}}$ using only the genes in $\mathbb{G}$, using the ProMMT algorithm's probability model. To account for the lower efficiency of *in situ* sequencing, we divided the means $\mu_{g,k}$ by a factor of 50 and on each iteration resampled the expression levels of each cell according to a Poisson distribution with this mean. We then computed a score $S[\mathbb{G}]$ as the mean similarity of the new cluster assignments $k'_{c;\ \mathbb{G}}$ to the original clusters k_c, with cluster similarity defined by the depth of the last common ancestral node of the two clusters on the binary classification tree.

The search was performed using a greedy algorithm, initializing $\mathbb{G}$ as an empty set. On each iteration, the algorithm computes the score increment $S[\mathbb{G} \cup g] - s[\mathbb{G}]$ that would be obtained by adding each gene g not currently in $\mathbb{G}$, and then adding the best gene. After this, it computes for each gene g currently in $\mathbb{G}$, a “gene value” $s[\mathbb{G}] - S[\mathbb{G} \setminus g]$, which measures how much the score would decrease if this gene was removed from the panel. Note that the value of any gene will decrease as the gene set grows larger, since genes will contain redundant information. If the value of any gene is negative on a given iteration, the gene with the most negative value was removed from $\mathbb{G}$. (A negative score means that retaining this gene in the set does more harm than good, which is possible since the Poisson resampling means genes whose expression provides no information will only contribute noise). The algorithm was run for 100 iterations.

After performing our mapping experiments, we re-evaluated the contribution of all genes to cell typing post hoc. We found that performance was improved by discarding Vsnl1, and was made no worse by discarding a further six (Supplementary Figure S19). We conclude that detecting more genes would not have been helpful, as genes whose expression is close to equal between classes only add noise to the classification problem.

Data analysis

Data was analyzed with a suite of custom software for image processing, gene calling, and cell calling. All code was written in MATLAB, and is freely available at https://github.com/kdharris101/iss.

In situ sequencing occurs in 5 rounds, each of which involves chemical processing followed by multispectral imaging of the tissue sample. Because the tissue sample is generally too large for a single camera image, imaging occurs in overlapping tiles. In each tile, a stack of 7 images covering 10 μm in depth were taken for each color, and flattened into 2D using an extended depth of focus algorithm [22]. The data therefore consists of a set of images

$$I_{R,C,T}(\mathbf{x})$$

Here I gives the pixel intensity for sequencing round R, color channel C, tile T, and pixel coordinates $\mathbf{x}$ within this tile. On each round, we have six images: a DAPI image; an anchor image that detects every sequenced RCP; and four images to detect individual bases in a position defined for that round. The processing pipeline to identify detected genes comprises several steps: initial registration; spot detection and fine registration; crosstalk compensation; and gene calling. These analyses proceed without ever “stitching” all the tiles into a single large image; this approach allows processing of very large datasets on computers with limited memory, and also easily allows non-rigid alignments. Prior to the pipeline, all RCP images are filtered with a disk-shaped top-hat filter with radius 3 pixels (corresponding to 1 µm, the expected RCP size) and all DAPI images are filtered with a disk-shaped top-hat filter with radius of 24 pixels (8 µm, the expected nuclear size).

Initial registration

Image registration proceeds in two steps. In the first step, we align the anchor channel images for all rounds, and compute the offsets between neighboring tiles. This initial step therefore defines a global coordinate system for the entire tissue sample, by computing the information that would be required to stitch the tiles together (although we never in fact create this global image array). In this initial step, non-linear registration is important, for example because the specimen might not lie flat under the microscope. The degree of nonlinear warping is small within a tile, but can amass to several pixels' shift across the entire (1cm) image, which would compromise the sequencing protocol if not properly accounted for. To solve this problem, we allow the shifts, scales, and rotations of each tile to the global coordinate system to differ, allowing nonlinearities at the global level.

Because we use a square tiling strategy, each tile may have up to four "neighbors": other tiles with which it has a region of substantial overlap. We denote the set of neighboring tile pairs as $\mathfrak{N}$. As the same tile configuration is used for each round, the neighbor relationships between tiles will not vary across rounds, even if a single RCP spot may occupy different tiles on different rounds.

We first align all tiles using the anchor channel on a "reference round" R_R (2 for the current analyses), which we refer to as the "reference image" for each tile. To align the reference images, we loop over all pairs of neighboring tiles, and compute an offset, using phase correlation to register the overlapping regions of the top hat-filtered reference images of these two tiles. The result is a shift vector $\mathbf{\Delta}_{T_1,T_2}$ for every pair of neighboring tiles T_1 and T_2 that specifies the x and y offsets of tile T_2 relative to tile T_1.

We next define single global coordinate system by finding the coordinate origin $\mathbf{X}_T$ for each tile T. Note however that this problem is overdetermined as there are more neighbor pairs than there are tiles. We therefore compute the offsets by minimizing the loss function [23,24].

$$L = \sum_{(T_1, T_2) \in \mathfrak{N}} \left| \mathbf{X}_{T_1} - \mathbf{X}_{T_2} - \mathbf{\Delta}_{T_1, T_2} \right|^2$$

Differentiating this loss function with respect to $\mathbf{X}_T$ yields a set of simultaneous linear equations, whose solution yields the origins of each tile on the reference round.

The results of this step suffice to define a global coordinate system, but do not provide pixel-level alignment of images from multiple color channels on multiple rounds, due to the occurrence of chromatic aberration and small rotational or non-rigid shifts. The latter will be dealt with in the next step, through point-cloud registration.

Spot detection and fine registration

The second processing step detects spots in all images, performs fine alignment of color channels and sequencing rounds, and computes for each spot a position in global coordinates and an intensity vector summarizing that spot's detected fluorescence in each round and channel.

The most intricate part of this step is fine image registration. Even though the same tile layout is used for all sequencing rounds, the precise positions of the tiles may differ due to slight shifts in the placement and rotation of the sample. Thus, a single spot might be found on different tiles in different sequencing rounds. Furthermore, due to chromatic aberration a spot may be in slightly different positions (although not different tiles) in different color channels. Because most spots are only a few pixels in size, even a one-pixel registration error can compromise accurate reads.

Spots first are detected in the reference images (anchor channel, reference round). For each tile, spots are detected as local maxima of the top hat-filtered image exceeding a fixed detection threshold. A global coordinate is defined for each of these spots using the initial registration described above. In regions where tiles overlap, duplicate spots are rejected by keeping only spots which are closer in global coordinates to the center of their original tile than to any other.

Next, spot positions are detected in images from all sequencing rounds, and all color channels. These are used to align each round and color channel to the anchor round reference channel, using point-cloud registration. Specifically, we fit an affine transformation from each reference image, to the images of the corresponding tile for all rounds and color channels, using the iterative-closest point (ICP) algorithm with matches further than 3 pixels away excluded. These affine transformations can include shifts, scalings, rotations and shears, but we did not find it necessary to introduce nonlinear warping transformations within tiles (Supplementary Figure S6E; nonlinear transformations can still occur globally by variation of the affine transformation across tiles). As the ICP algorithm is highly sensitive to local maxima, it is initialized from a shift transformation computed by phase correlation of anchor channel images. When spots are located on neighboring tiles on different rounds, the corresponding images are again registered with ICP.

Finally, an intensity vector is computed for each spot, by reading the intensity from the aligned coordinate of each top hat-filtered image. Although the point-cloud registration yields subpixel alignment we did not apply subpixel interpolation to the images, instead filtering with a radius 1 disk filter to allow images to be detected after subpixel shifts.

Crosstalk compensation and gene-calling

The last step associating spots to genes consists of transforming the intensity vectors to gene identities.

An important consideration in this stage is that crosstalk can occur between color channels. Some crosstalk may occur due to optical bleedthrough; additional crosstalk can occur due to chemical cross-reactivity of probes. The precise degree of crosstalk can vary between sequencing rounds, but tends to be constant within a round. It is therefore possible to largely compensate for this crosstalk by learning the precise amount of crosstalk between each pair of color channels on each round.

To estimate the crosstalk present on a given round *r*, we first collect a set of 4-dimensional vectors $\mathbf{v}_{s,r}$ containing the intensity in each color channel of all well-isolated spots *s*. Only well-isolated spots are used to ensure that crosstalk estimation is not affected by spatial overlap of spots corresponding to different genes; a spot is defined as well-isolated if the reference image intensity averaged over an annular region (2-7 pixel radius) around the spot is less than a threshold value (60 for current analyses, applied to 16-bit images after top-hat filtering). Crosstalk is then estimated by running a scaled k-means algorithm [25] on these vectors, which finds a set of four vectors $\mathbf{c}_{b,r}$ (*b* refers to one of the four base possibilities in round *r*), such that the error function $\sum_s \min_{\lambda_s, b(s)} |\mathbf{v}_{s,r} - \lambda_s \mathbf{c}_{b(s),r}|^2$ is minimized; in other words, it finds for each round *r* the four intensity vectors $\mathbf{c}_{b,r}$ such that each well-isolated spot on round *r* is close to a scaled version of one of them.

Finally, we associate each spot with a gene using the codebook defined by the probe barcodes. For each probe *p* with barcode $b_1^p, \ldots b_5^p$, we concatenate the corresponding crosstalk vectors into a 20-dimensional vector $\left[\mathbf{c}_{b_{1,1}}^p, \mathbf{c}_{b_{2,2}}^p, \mathbf{c}_{b_{3,3}}^p, \mathbf{c}_{b_{4,4}}^p, \mathbf{c}_{b_{5,5}}^p\right]$. Each spot is called as belonging to the probe for which this vector is best matches the spot's 20-dimensional intensity vector, as measured by normalized dot-product (i.e. the cosine angle between the measured intensity vector and crosstalk-compensated code vector). Spots whose cosine angles fall below a threshold value are taken to represent misreads (for example due to background fluorescence) and discarded. The threshold value (0.9 for the current analyses) was chosen manually as a value below which reads appeared not matching the known genomic composition of CA1 interneurons established by prior scRNA-seq; 63% of reads passed the threshold in current experiments.

Cell calling

To assign cells to classes, we used a probabilistic approach. We start with a model that predicts the probability of any configuration of RNA detection spots, given the class of every cell. We then use Bayes' theorem to estimate the probability for each cell to belong to each class, given the observed RNA spot configuration. To do this, we must also estimate the probability distributions of other "hidden variables", such as the cell responsible for each RNA detection, and the detection efficiency of each gene. The current algorithm however does not estimate the mean expression level of each gene in each cell class; instead it relies on these means being defined by previous analysis of scRNA-seq data, where higher efficiency and larger cell counts lead to more accurate estimates of these parameters.

Notation and preliminaries

Cellular RNA counts can be accurately modelled by a negative binomial distribution [26,27]. The negative binomial is a better model of RNA counts than the simpler Poisson distribution, as it has a larger variance, that matches measured fluctuations in gene expression. We parametrize the negative binomial distribution by its mean μ and a dispersion parameter r for which a value of $r = 2$ fits CA1 neurons well (Ref. [6], Supplementary Figure S2). Note that parameterizing the negative binomial by its mean is different to the usual parameterization in terms of success probability. In terms of these parameters, the probability distribution is:

$$NB(k; r, \mu) = \binom{k + r - 1}{k}\left(\frac{\mu}{\mu + r}\right)^{k}\left(\frac{r}{\mu + r}\right)^{r}$$

The notation $\binom{n}{r}$ denotes combinations: $\binom{n}{r} = \frac{n!}{r!(n - r)!}$.

Our algorithm will take advantage of the fact that a negative binomial distribution can be defined as a Poisson distribution whose mean is itself random following a gamma distribution. We parametrize the gamma distribution by a shape r and rate β, with probability density function:

$$Gamma(x; r, \beta) = \frac{\beta^{r}}{\Gamma\ (r)}x^{r - 1}e^{-\beta x}$$

Recall that if $x \sim Gamma(x;\ r,\ \beta)$ then $E(x) = r/\beta$, $E(\log\ x) = \psi(r) - \log(\beta)$ where $\psi(r)$ is the digamma function, and $\Lambda\ x \sim Gamma\left(x; r, \frac{\beta}{\Lambda}\right)$, for any $\Lambda > 0$. The relationship between the gamma, Poisson, and negative binomial distributions is as follows: if $x \sim Poisson(\lambda)$ and $\lambda \sim Gamma(r, r/\mu)$, then $x \sim NB(r, \mu)$.

We will represent the results of an *in situ* sequencing experiment via the location $\mathbf{x}_s$ and decoded gene g_s of each detected RNA spot s. We represent the cell of origin of an RNA spot s as $c(s)$, and define an indicator variable $z_{s,c}$ to be 1 if spot s arose from cell c and 0 otherwise: $z_{s,s(c)} = 1$. Similarly, we denote by $k(c)$ the cell class of cell c, and define an indicator variable $\zeta_{c,k}$ to be 1 if cell c belongs to class k and 0 otherwise: $\zeta_{c,k(c)} = 1$. Note that $\sum_c z_{s,c} = 1$ for all s, and $\sum_k \zeta_{c,k} = 1$ for all c. The letters z and ζ written without subscripts refer to the entire matrices of these indicator variables.

Assigning spots to cells

Most RNAs are detected within somas, the cytoplasm near cell nuclei, but many are also located more distal from the soma. Assigning RNA spots to their cells of origin is therefore a non-trivial problem. We do this using a probabilistic framework, allowing for the fact that a spot's location does not identify its parent cell with complete certainty.

We detect cell nuclei using DAPI staining, and the DAPI image is segmented to reveal an approximately circular region outlining each cell. In our model, spots inside this region are highly likely (but still not absolutely certain) to arise from the cell; and the probability of a spot arising from the cell decays progressively with distance from the DAPI region.

To formalize this mathematically, denote the centroid of cell *c*'s DAPI region as $\mathbf{x}_c$, and an indicator function $I_c(\mathbf{x})$ to be 1 if point **x** lies within the DAPI region. We define a function measuring the distance from a point **x** to a cell *c* as:

$$D_C(\mathbf{x}) = \frac{\left|\mathbf{x} - \mathbf{x}_c\right|^2}{2\bar{r}^2} + \log\left(2\pi\bar{r}^2\right) - bI_c(\mathbf{x})$$

Here r_0 is the mean radius of the DAPI region over all cells. Note that the first two terms define the negative log of a normalized Gaussian density of radius r_0. The third term produces a bias toward identifying a point inside the DAPI region with its cell of origin, with the parameter *b* taking the value 3 for our current analyses; this value was chosen manually after inspecting the assignment of gene reads to cells (as in Figure 2A), to confirm that reads both inside and outside the DAPI regions matched the choices that a human operator with knowledge of this cell system would make.

Later calculations will require a measure of each cell's normalized area:

$$A_c = \int e^{-D_C(\mathbf{x})} d\mathbf{x}$$

If *b* were equal to 0, A_c would be 1 for all cells, due to the normalization of the log-density D_c. Numerical computation of the integral would be time-consuming due to the large number of cells present, and we therefore use an approximation assuming each cell is circular. If cell *c* is approximately circular with radius r_c, a simple integration shows that

$$A_c \approx e^b + e^{-r_c^2/2\bar{r}^2}\left(1 - e^b\right)$$

Not all spots can be identified with cells; RNAs located in cellular processes are so far from somata it is impossible to identify the soma of origin; and others arise from technical misreads. To account for these, we add an additional source of spots corresponding to a uniform density ρ_0, which equals 10^{-5} misreads/pixel for current analyses:

$$D_0(\mathbf{x}) = -\log\rho_0$$

Including this misread density allows the algorithm to automatically discard any rare gene misreads that nevertheless passed the cosine distance threshold (for example due to off-target probe binding). The value of 10^{-5} was chosen based on visual estimates of the number of reads seen not matching transcriptomic classes established by scRNA-seq: approximately 1 misread every 20 cells.

Probability model

The number of counts of a gene *g* in a cell *c* can be modelled as $x_{gc} \sim NB(r, \mu_{g,k(c)})$, where $k(c)$ represents the cell class to which cell *c* belongs, $\mu_{g,k}$ represents the mean RNA count of gene *g* in cell class *k*, and *r* is a parameter, for which the value of 2 provides a good fit [6]. Note that in this manuscript we parameterize the negative binomial by *r* and its mean μ, rather than the probability parameter $p = \mu/(r + \mu)$.

For our current purposes, however, a model for each cell's RNA counts is not sufficient: we need a probability distribution for not just the number of spots, but also their locations. This kind of probability distribution is known as a *spatial point process*[28].

The best-characterized spatial point process is the (inhomogeneous) *Poisson process.* A Poisson process is parametrized by an intensity function $\lambda(\mathbf{x})$, which measures the density of points expected to be found at every location **x**. Given an intensity function, the Poisson process assigns a spot configuration $\{\mathbf{x}_s : s = 1 \ldots S\}$ the log probability density:

$$\log P\left(\boldsymbol{x}_s \middle| \lambda\right) = -\int \lambda(\boldsymbol{x}) d\boldsymbol{x} + \sum_s \log \lambda\left(\boldsymbol{x}_s\right)$$

A key property of the Poisson process is that the total number of points in any region of space follows a Poisson distribution, with mean equal to the integral of the intensity function in this region. Thus, a Poisson process is not itself sufficient to model negative-binomial RNA counts.

To model the number and spatial locations of the RNA spots produced by a given cell, we take advantage of the fact that a negative binomial distribution arises when the mean of a Poisson distribution is itself random, following a gamma distribution. Specifically, if $x \sim Poisson(\lambda)$ and $\lambda \sim Gamma(r, r/\mu)$, then $x \sim NB(r, \mu)$.

We model the distribution of RNA spots of gene *g* arising from cell *c* as a Poisson process with intensity function

$$\lambda_{g,c}(\boldsymbol{x}) = \mu_{g,k(c)} e^{-D_c(\boldsymbol{x})} \gamma_{g,c} \eta_g$$

Here, $k(c)$ represents the class of cell *c*; $\mu_{g,k}$ represents the mean expression level of gene *g* in cell class *k* as determined by scRNA-seq; $D_c(\mathbf{x})$ is the function measuring the distance of point *x* from cell *c* (see above); and $\gamma_{g,c}$ represents a gamma-distributed scale factor for each cell and gene, representing fluctuations in gene expression levels that cause the total expression level to follow a negative binomial rather than Poisson distribution. In our model, $\gamma_{g,c} \sim Gamma(r, 1)$, where the shape parameter r takes the value 2 to ensure the negative binomial distribution has correct dispersion. Finally, η_g represents the efficiency of *in situ* sequencing of gene *g* relative to single-cell sequencing. Because we do not know the efficiencies *a priori*, we also model the efficiency of each gene probabilistically: $\eta_g \sim Gamma(r, \eta_0)$, where the expected efficiency η_0 takes the value 0.2 for current analyses, and we use a shape parameter $r = 20$. This prior distribution allowed the efficiency of each gene to be estimated for each experiment, allowing the algorithm to account for gene-specific technical fluctuations in efficiency. The mean value of 0.2 was chosen based on previous estimates of the efficiency of this method, but is “uninformative”: the large prior variance $r = 20$ ensures that the effect of this prior mean is quickly overridden by data.

To write the formula for the full probability distribution, we use the “indicator variables” $z_{s,c}$ which is 1 if spot s arose from cell c and 0 otherwise; and $\zeta_{c,k}$ which is 1 if cell c belongs to class k (i.e. if $k = k(c)$) and 0 otherwise. We define $\boldsymbol{\pi}_{\mathrm{k}}$ is the prior probability of a cell to belong in class k (Supplementary Table S4). Then we have

$$\log P(\mathbf{x}, g, z, \zeta, \gamma, \eta) = -\sum_{g,c,k} \zeta_{c,k} \int \mu_{g,k} e^{-D_c(\mathbf{x})} \gamma_{c,g} \eta_g d\mathbf{x} + \sum_{s,c,k} z_{s,c} \zeta_{c,k} \log(\mu_{g,k} e^{-D_c(\mathbf{x}_s)} \gamma_{c,g_s} \eta_g) \\ + \sum_{g,c} \log Gamma(\gamma_{g,c} | r, r) + \sum_g \log Gamma(\eta_g \big| r, r/\eta_0) + \sum_{c,k} \zeta_{c,k} \log \pi_k$$

Defining $A_c = \int e^{-D_c(\mathbf{x})} d\mathbf{x}$, this simplifies to

$$\begin{aligned} \log P(\mathbf{x}, g, z, \zeta, \gamma, \eta) \\ = -\sum_{g,c,k} \zeta_{c,k} \mu_{g,k} A_c \gamma_{c,g} \eta_g \\ + \sum_{s,c} z_{s,c} \left[-D_c(\mathbf{x}_s) + \log \gamma_{c,g_s} + \log \eta_{g_s} + \sum_{\mathrm{k}} \zeta_{c,k} \log \mu_{g_s,k} \right] \\ + \sum_{g,c} \log Gamma\left(\gamma_{g,c} | r, r\right) + \sum_g \log Gamma\left(\eta_g \middle| r_\eta, r_\eta/\eta_0\right) + \sum_{c,k} \zeta_{c,k} \log \pi_k \end{aligned} \tag{1}$$

Variational Bayes approximation

We would like to obtain the posterior distribution of the cell classes given the data: $Prob(\zeta|\mathbf{x}, g)$. Direct application of Bayes’ theorem is analytically intractable, and we therefore employ the mean-field variational Bayes approximation, a common method in Bayesian analysis that is conceptually similar to the Expectation-Maximization algorithm of classical statistics [29]. In this approach, we approximate the posterior distribution of the unobserved variables by a product $Prob(z,\zeta,\gamma,\eta|\mathbf{x},g) \approx q(\zeta,\gamma)q(z)q(\eta)$, and alternate estimating the three functions q while holding the others fixed. On each step, log q is estimated as the expectation of the log total probability over the other unobserved variables, plus a normalizing constant [46]

We group the variables ζ and γ together as the appropriate values of $\gamma_{c,g}$ for a cell c will depend on the class of that cell. To compute $q_1(\zeta,\gamma)$ we first see that

$$E_{z,\eta} \log P(\mathbf{x}, g, z, \zeta, \gamma, \eta) = -\sum_{g,c,k} \zeta_{c,k} \mu_{g,k} A_c \gamma_{c,g} \overline{\eta_g} + \sum_{s,c} \overline{z_{s,c}} \left[\log \gamma_{c,g_s} + \sum_{\mathrm{k}} \zeta_{c,k} \log \mu_{g_s,k} \right] \\ + \sum_{g,c} log\ Gamma(\gamma_{g,c} | r, r) + \sum_{c,k} \zeta_{c,k} \log \pi_k + const$$

Here are overbar represents the expectation of a unobserved variable with respect to its current q distribution, and *const* collects terms that do not depend on ζ or γ. Writing $N_{c,g}$ for the total number of spots of gene g assigned to cell c, i.e. $N_{c,g} = \sum_{s:g_s = g} z_{s,c}$, and remembering that $\sum_k \zeta_{c,k} = 1$ for all c, we can switch the sum over spots in the second term to a sum over genes:

$$\log q(\zeta, \gamma) = \sum_{g,c,k} \zeta_{c,k}[-\mu_{g,k} A_c \gamma_{c,g} \overline{\eta_g} + \overline{N_{g,c}} \log(\gamma_{c,g} \mu_{g,k}) + \log Gamma(\gamma_{g,c} | r, r)] + \sum_{c,k} \zeta_{c,k} \log \pi_k + const$$

We next factorize this joint probability distribution $q1(\zeta, \gamma)$ as a marginal and a conditional: $q(\zeta, \gamma) = q(\zeta)q(\gamma|\zeta)$. To obtain $q(\zeta)$ we could integrate $\int q(\gamma|\zeta)d\gamma$, and normalize to a probability distribution. In practice, however, this is unnecessary. We can see by inspection that for any g and c, the summand of the top term is the log probability of a gamma-Poisson mixture, which defines a negative binomial when integrated over $\gamma_{g,c}$. We therefore have:

$$\log q(\zeta) = \sum_{g,c,k} \zeta_{c,k} \left(\log NB\left(\overline{N_{g,c}}; r, \mu_{g,k} A_c \overline{\eta_g} \right) + \log \pi_k \right)$$

Rewriting this in terms of the class assignment variables $k(c)$ we have:

$$q(k(c) = k) \propto \pi_k \prod_g NB\left(\overline{N_{g,c}}; r, \mu_{g,k} A_c \overline{\eta_g} \right) \quad (2)$$

For each cell c, the estimated class probabilities are thus those obtained observing $\overline{N_{g,c}}$ of copies of each gene g (i.e. the expected number assigned to the cell given the current distribution of spot assignments), under a negative binomial distribution of mean $\mu_{g,k} A_c \overline{\eta_g}$ (i.e. the scRNA-seq means scaled by the current estimate of *in situ* efficiency and cell area).

To specify the conditional distribution $q(\gamma|\zeta)$, we must obtain for each cell c and gene g a probability distribution for $\gamma_{c,g}$ conditional on each possible cluster assignment $k(c)$ for that cell. Some manipulation shows that

$$q\left(\gamma_{g,c} \middle| k(c)\right) = Gamma\left(\gamma_{g,c}; r + \overline{N_{g,c}}, r + \mu_{g,k(c)} A_c \overline{\eta_g}\right) \quad (3)$$

Thus, for each possible class assignment $k(c)$, the scale factor $\gamma_{g,c}$ follows a gamma distribution, whose mean approaches $\overline{N_{g,c}} / \left(\mu_{g,k(c)} A_c \overline{\eta_g} \right)$, i.e. the ratio between the number of reads of each gene assigned to that cell, to the number predicted from scRNA-seq counts, cell area, and estimated efficiency.

We now turn to the estimated distribution for the spot assignments, $q(z)$. From equation (1) we see that:

$$E_{\zeta,\gamma,\eta} \log P(\mathbf{x}, g, z, \zeta, \gamma, \eta) = \sum_{s,c} z_{s,c} \left[-D_c(\mathbf{x}_s) + \sum_k \overline{\zeta_{c,k}} \log \mu_{g_s,k} + \overline{\log \gamma_{g,c}} \right] + const$$

Rewriting this in terms of the assignment variables *c*(*s*) we have:

$$q(c(s) = c) \propto \exp\left[-D_c(\mathbf{x}_s) + \overline{\log \gamma_{g,c}} + \sum_k \overline{\zeta_{\mathrm{c,k}}} \log \mu_{g_s,k}\right] \tag{4}$$

The expectation $\overline{\zeta_{c,k}}$ is simply the probability *q*(*k*(*c*) = *k*), and we can compute $\overline{\log \gamma_{g,c}} = \sum_k q(k(c) = k) E_{q\left(\gamma_{g,c} | k(c)\right)}\left[\log \gamma_{g,c}\right]$ by plugging the parameters from equation (3) into the formula for the expected log of a gamma variate. This shows that the probability of assigning a spot to a given cell will be large when the spot is close to the cell and the likely class assignments of that cell have high expression of the gene.

Finally, we must compute *q*(*η*), the distribution of *in situ* efficiency parameters for each gene. From equation (1) we see that:

$$E_{\zeta,\gamma,z} \log P(\mathbf{x}, g, z, \zeta, \gamma, \eta) = -\sum_{g,c,k} \mu_{g,k} A_c \overline{\gamma_{c,g}} \eta_g + \sum_s \log \eta_{g_s} + \sum_g \log Gamma\left(\eta_g \middle| r_\eta, r_\eta / \eta_0\right)$$

We therefore have $q(\eta) = \prod_g q(\eta_g)$, and a quick calculation shows that:

$$q(\eta) = Gamma\left(r_\eta + N_g, r_\eta / \eta_0 + \sum_{c,k} \mu_{g,k} A c \overline{\gamma_{c,g}}\right) \tag{5}$$

Thus, the efficiency factor for gene *g* follows a gamma distribution whose mean approaches $N_g / \sum_{c,k} \mu_{g,k} A_c \overline{\gamma_{c,g}}$, the ratio of the total number of reads of that gene to the summed predictions of each cells scRNA-seq, area, and scale factor.

Regularizing the model of gene expression

Although Bayesian approaches provide optimal answers when the underlying probability models are accurate, they can be highly sensitive to errors that are not captured by the probability model. For example, if expression of gene *g* in cell type *k* were modelled by a negative binomial distribution with mean 0, detecting a single copy of gene *g* would make it impossible for the cell to be classified as class *k*, even if expression of all other genes matched class *k* perfectly. To model the fact that such detections might occur through technical errors, we therefore take the mean expression parameter $\mu_{g,k}$ to be the value obtained by scRNA-seq plus a regularization parameter *ν*, set to 10^{-3} in the current analyses. Experimenting with different values of this parameter we found its exact value had little effect provided it was non-zero, and therefore took an extremely low value of 10^{-3} reads/cell.

The present method does not aim to classify all cell types, and only genes targeting neurons have been included in the probe set. Consequently, many cells detected by DAPI have zero or few detected RNAs. To account for these cells, we have included an additional cell class “Zero”, with $\mu_{g,0} = \nu$ for all *g*.

Optimizing for speed

In principle, the algorithm allows computing the probability of every RNA spot to belong to every cell. This would be computationally very slow; furthermore, most of these potential matches are impossible as the cells are simply too far away from the spots. We therefore restrict the search for the parent cell of each spot to only its three closest neighbors

Algorithm summary

The algorithm is summarized in the following pseudocode:

```
% Initialize variables:
Compute regularized mean expression $\mu_{g,k}$ from scRNA-seq data including “zero” class
Compute distance parameters $D_C(\boldsymbol{x}_S)$ for three closest neighbors and misread density
Compute normalized area of each cell $A_C$
Initialize gene scale factors $\eta_g$ to have mean 0.2
Initialize cell scale factors $\gamma_{c,g|k}$ to have mean 1
Assign each spot to closest neighbor with probability 1
% main loop
Repeat until convergence:
  Compute expected RNA count in each cell $\overline{N_{g,c}}$
  Compute cell class probabilities using equation 2
  Compute gamma distribution parameters for scale factors $\gamma_{c,g|k}$ using equation 3
  Compute gamma distribution parameters for in situ efficiencies $\eta_g$ using equation 5
  Compute spot assignment probabilities using equation 4
```

The algorithm is determined to have converged when the spot assignments have stopped changing. Specifically, for every spot we compute the amount its assignment probabilities $\overline{z_{s,c}}$ have changed since the last iteration, using the l_∞ norm: $\max_c \left| \overline{z_{s,c}} - \overline{z_{s,c,OLD}} \right|$. When the mean value of this across cells is lower than a tolerance threshold (0.02 for present analyses), the loop terminates.

Statistics

The data presented in the study was generated from three independent experiments on 14 mouse brain sections from one animal. The computational method presented in this study is based on Bayes’ theorem and we also included a regularization parameter to avoid spurious cell type assignment. Although we did not conduct any statistical testing between experiments, we observed good correlation between sections and hemispheres.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

Acknowledgments

We thank Peter Somogyi, Matteo Carandini, Sten Linnarsson, Markus Hilscher, Nicoletta Kessaris and Lorenza Magno for valuable discussions. We thank Kasper Karlsson for providing scRNA-seq reads for *Cxcl14* gene. This work was supported by grants from the Wellcome Trust (108726, to KDH, JHL, and MN), Chan-Zuckerberg Initiative (182811 to KDH), the Swedish Research Council (2016-03645 to MN), Knut och Alice Wallenbergs Stiftelse (to MN) and Familjen Erling-Perssons Stiftelse (to MN).

References

1. Lein E, Borm LE, Linnarsson S. The promise of spatial transcriptomics for neuroscience in the era of molecular cell typing. Science. 2017; 358:64–69. [PubMed: 28983044]
2. Zeisel A, et al. Brain structure. Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq. Science. 2015; 347:1138–1142. [PubMed: 25700174]
3. Tasic B, et al. Adult mouse cortical cell taxonomy revealed by single cell transcriptomics. Nat Neurosci. 2016; 19:335–346. [PubMed: 26727548]
4. Cembrowski MS, Wang L, Sugino K, Shields BC, Spruston N. Hipposeq: a comprehensive RNA-seq database of gene expression in hippocampal principal neurons. eLife. 2016; 5:e14997. [PubMed: 27113915]
5. Paul A, et al. Transcriptional Architecture of Synaptic Communication Delineates GABAergic Neuron Identity. Cell. 2017; 171:522–539.e20. [PubMed: 28942923]
6. Harris KD, et al. Classes and continua of hippocampal CA1 inhibitory neurons revealed by single-cell transcriptomics. PLoS Biol. 2018; 16:e2006387. [PubMed: 29912866]
7. Tasic B, et al. Shared and distinct transcriptomic cell types across neocortical areas. Nature. 2018; 563:72. [PubMed: 30382198]
8. Zeisel A, et al. Molecular Architecture of the Mouse Nervous System. Cell. 2018; 174:999–1014.e22. [PubMed: 30096314]
9. Shah S, Lubeck E, Zhou W, Cai L. *In Situ* Transcription Profiling of Single Cells Reveals Spatial Organization of Cells in the Mouse Hippocampus. Neuron. 2016; 92:342–357. [PubMed: 27764670]
10. Cembrowski MS, Spruston N. Integrating Results across Methodologies Is Essential for Producing Robust Neuronal Taxonomies. Neuron. 2017; 94:747–751.e1. [PubMed: 28521129]
11. Shah S, Lubeck E, Zhou W, Cai L. seqFISH Accurately Detects Transcripts in Single Cells and Reveals Robust Spatial Organization in the Hippocampus. Neuron. 2017; 94:752–758.e1. [PubMed: 28521130]
12. Freund TF, Buzsaki G. Interneurons of the hippocampus. Hippocampus. 1996; 6:347–470. [PubMed: 8915675]
13. Pelkey KA, et al. Hippocampal GABAergic Inhibitory Interneurons. Physiol Rev. 2017; 97:1619–1747. [PubMed: 28954853]
14. Somogyi, P. Hippocampus: intrinsic organizationHandbook of Brain Microcircuits. Shepherd, GM, Grillner, S, editors. Oxford University Press; 2010.
15. Wang X, et al. Three-dimensional intact-tissue sequencing of single-cell transcriptional states. Science. 2018; 361
16. Moffitt JR, et al. Molecular, spatial and functional single-cell profiling of the hypothalamic preoptic region. Science. 2018; :eaau5324.doi: 10.1126/science.aau5324 [PubMed: 30385464]
17. Codeluppi S, et al. Spatial organization of the somatosensory cortex revealed by osmFISH. Nat Methods. 2018; 15:932. [PubMed: 30377364]
18. Ke R, et al. *In situ* sequencing for RNA analysis in preserved tissue and cells. Nat Methods. 2013; 10:857–860. [PubMed: 23852452]
19. Lein ES, et al. Genome-wide atlas of gene expression in the adult mouse brain. Nature. 2007; 445:168–76. [PubMed: 17151600]
20. Eng C-HL, et al. Transcriptome-scale super-resolved imaging in tissues by RNA seqFISH+ Nature. 2019; 568:235. [PubMed: 30911168]

21. ChenK H, Boettiger AN, Moffitt JR, Wang S, Zhuang X. Spatially resolved, highly multiplexed RNA profiling in single cells. Science. 2015; 348:aaa6090. [PubMed: 25858977]

22. Pertuz S, Puig D, Garcia MA, Fusiello A. Generation of All-in-Focus Images by Noise-Robust Selective Fusion of Limited Depth-of-Field Images. IEEE Trans Image Process. 2013; 22:1242–1251. [PubMed: 23221820]

23. Hörl D, et al. BigStitcher: Reconstructing high-resolution image datasets of cleared and expanded samples. bioRxiv. 2018; doi: 10.1101/343954

24. Preibisch S, Saalfeld S, Tomancak P. Globally optimal stitching of tiled 3D microscopic image acquisitions. Bioinformatics. 2009; 25:1463–1465. [PubMed: 19346324]

25. Elad, M. Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing. Springer-Verlag; 2010.

26. Robinson MD, Smyth GK. Small-sample estimation of negative binomial dispersion, with applications to SAGE data. Biostatistics. 2008; 9:321–332. [PubMed: 17728317]

27. Lu J, Tomfohr JK, Kepler TB. Identifying differential expression in multiple SAGE libraries: an overdispersed log-linear model approach. BMC Bioinformatics. 2005; 6:165. [PubMed: 15987513]

28. Baddeley, A, Rubak, E, Turner, R. Spatial Point Patterns: Methodology and Applications with R. CRC Press; 2015.

29. Bishop, CM. Pattern Recognition and Machine Learning | Christopher Bishop | Springer. Springer verlag; 2006.

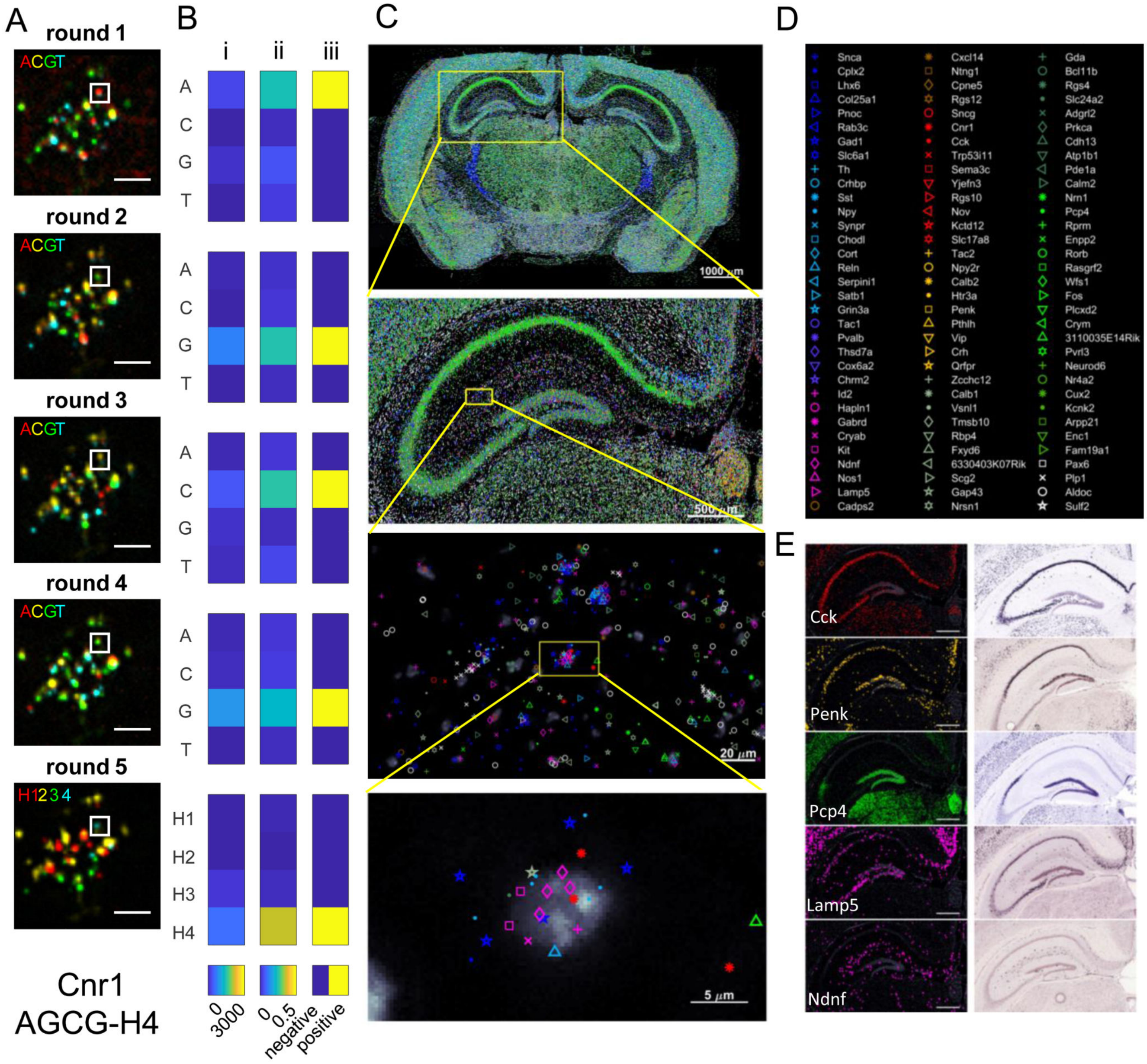

Figure 1. Detection of 99 genes in a mouse brain coronal section.

A) Pseudocolor images showing barcode sequencing readout for a region corresponding to one cell. Top to bottom, base-specific fluorophores in the four cycles of sequencing by ligation, and for the fifth cycle of barcode specific hybridization. The white square shows a single RCP of barcode AGCG-H4. Scale bars: 5 µm. **B)** Gene-calling for this RCP. Left: pseudocolor representation of raw fluorescence intensities; Middle, intensity after crosstalk compensation; Right, best fit barcode (AGCG-H4, encoding the gene *Cnr1*). **C)** Distribution of 99 genes at different zoom levels. From top to bottom: a complete coronal mouse brain section; left hippocampus; the border of stratum radiatum and stratum lacunosum moleculare; finally, zoom-in to reads for the cell whose raw fluorescence is shown in panel (A). **D)** Code symbols for the 99 marker genes. **E)** Comparison of the distribution of five markers in the hippocampus as determined by pciSeq (left column) with the distribution shown in the Allen Mouse Brain Atlas (right column). Scale bars: 500 μm. Similar results were observed in all 14 sections from three experiments.

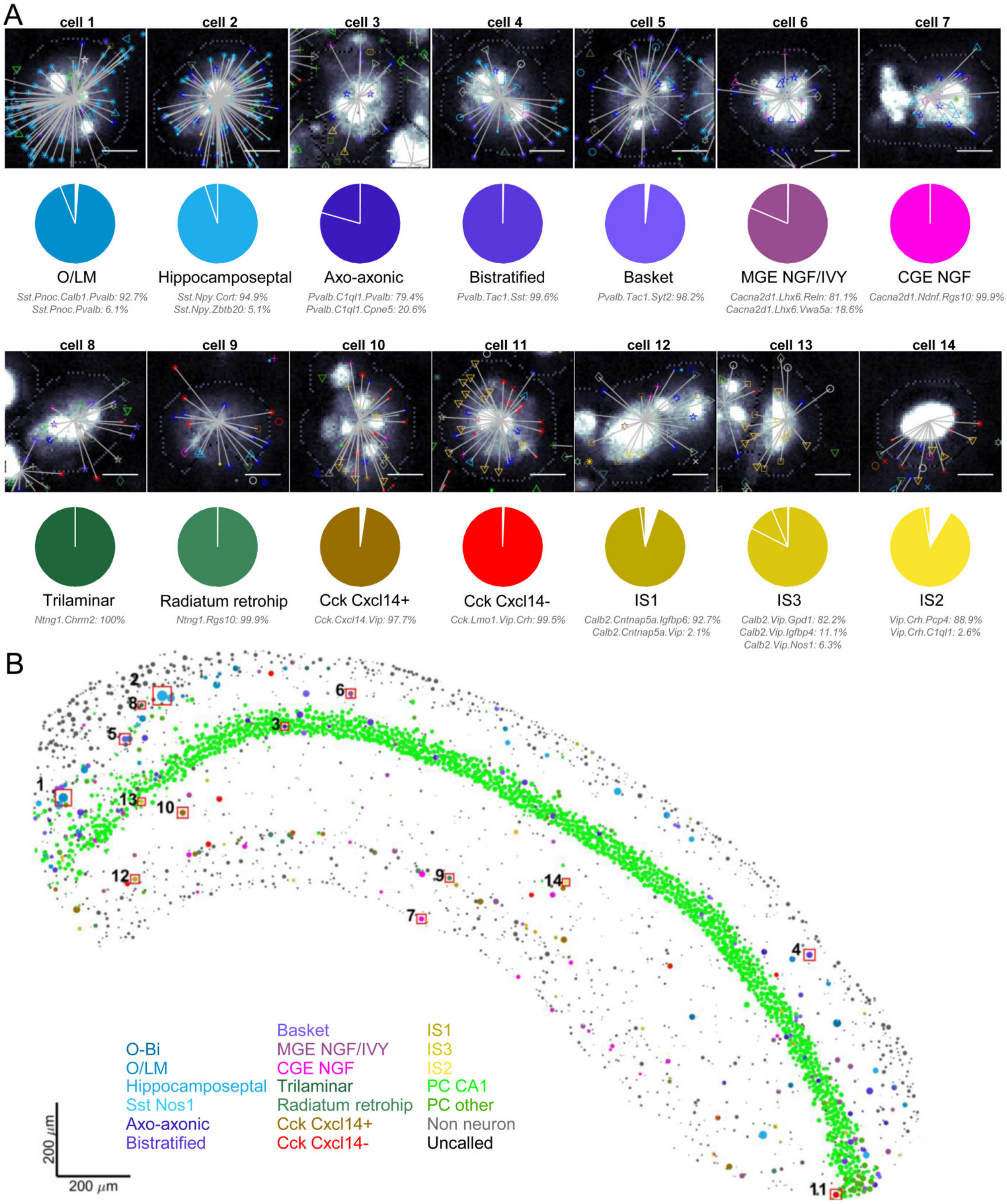

Figure 2. Cell type map of CA1 from an example experiment (experiment 4-3 right hemisphere).

A) Reads are assigned to cells, and cells to classes using a probability model based on scRNA-seq data. Top row: distribution and assignment of reads for fourteen example cells. Colored symbols indicate reads (color code as in Figure 1D). Grayscale background image indicates DAPI stain with watershed segmentation as dotted line. Straight lines join reads to the cell for which are assigned highest probability. Scale bars: 5 µm. Bottom row: pie charts showing probability distribution of each class for the same example cells. Colors indicate broad cell types; segments show probabilities for individual scRNA-seq clusters (named underneath). **B)** Spatial map of cell types across CA1. Cells are represented by pie charts, with radius proportional to square root of the number of reads assigned to the cell. Numbers identify the example cells in (A). Similar maps were obtained for all 28 hippocampus sections.

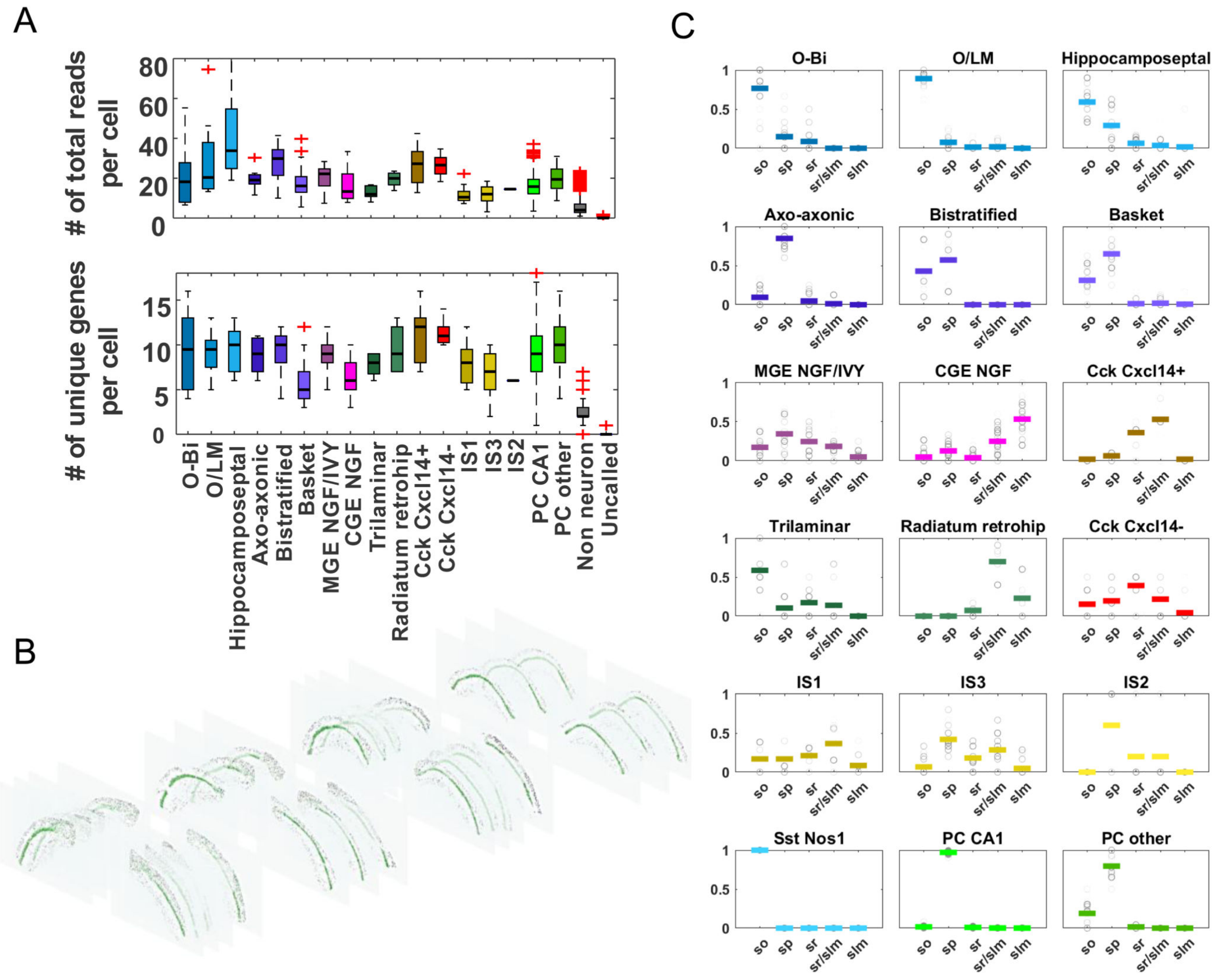

Figure 3. Validation of cell calling.

A) Box-and-whisker representation of total read count per cell of each type (top) and average number of unique genes per cell of each type (bottom) from n=3214 cells in the section shown in Figure 2B. Center line, median; box limits, upper and lower quartiles; whiskers, 1.5x interquartile range; points, outliers. **B)** 3D montage of cell calling results from all 14 sections processed. **C)** Fraction of each cell class found in each CA1 layer. Circles indicate means of a single experiment with gray level representing number of cells of that class in the experiment; colored lines denote grand mean over all 28 hippocampus sections. In each plot, the 5 x-axis positions represent layers: stratum oriens (so), stratum pyramidale (sp), stratum radiatum (sr), border of strata radiatum and lacunosum-moleculare (sr/slm), stratum lacunosum-moleculare (slm). MGE: medial ganglionic eminence. CGE: caudal ganglionic eminence. NGF: neurogliaform. IS: interneuron-selective cells.