# Supplementary Notes

Validation of astrocytes using staining images

We visualized the predicted astrocytes across visual cortex columns using DAPI and Nissl staining. In panel (a), we provide an overview picture of the 4 cortex columns (n=1597 cells) that have been imaged. Each color denotes a cell type that has been predicted by SVM. Then in panels (b-e), we focus on the staining patterns of individual cortex columns: columns 1-4 in (b-e). Within each panel among (b-e), the first row of each panel shows DAPI staining overlaid with cell type annotations (indicated by different colors). The second row shows Nissl staining of the same cells. Notice that astrocytes (indicated by magenta) show weak/no Nissl staining but clear DAPI staining.
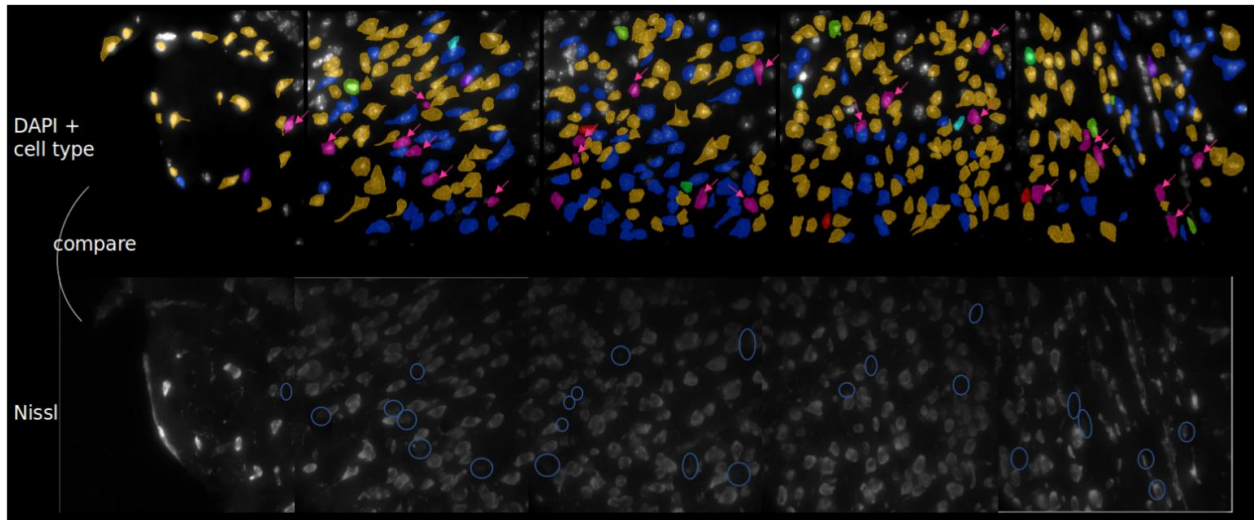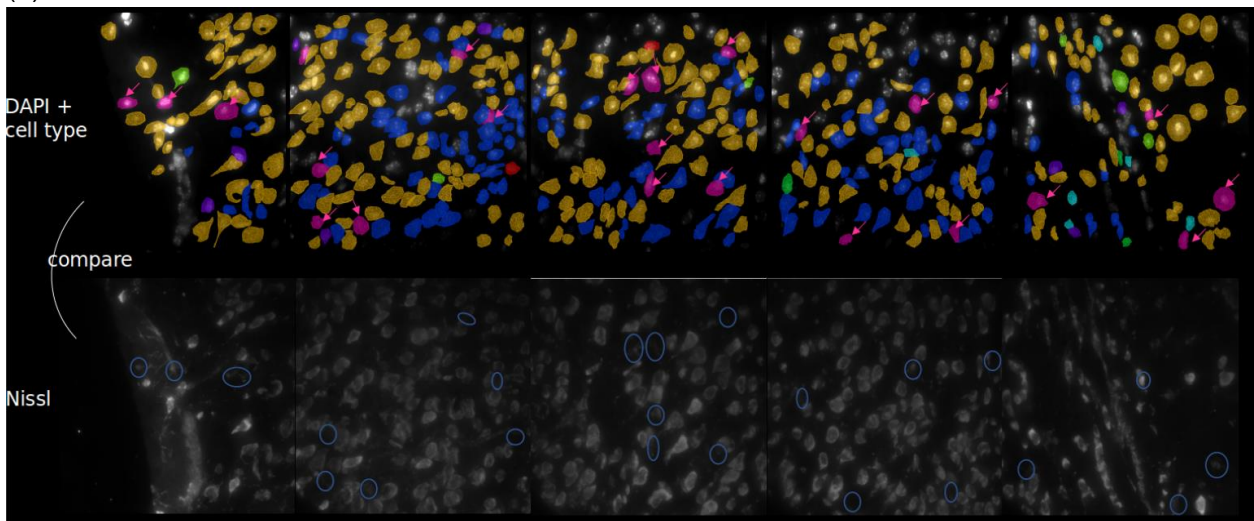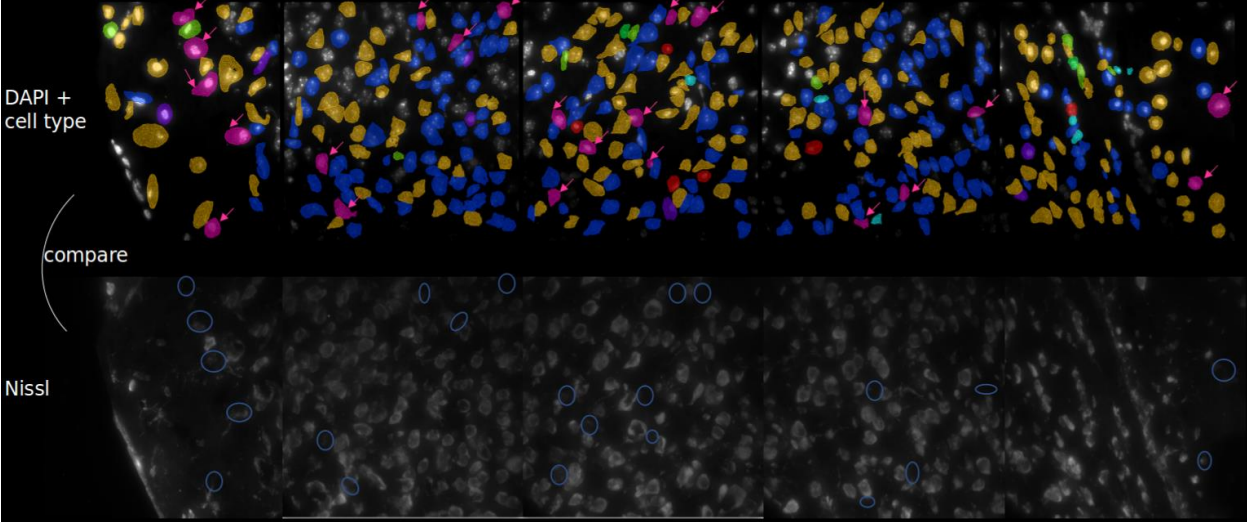
(a) Overview of cortex columns
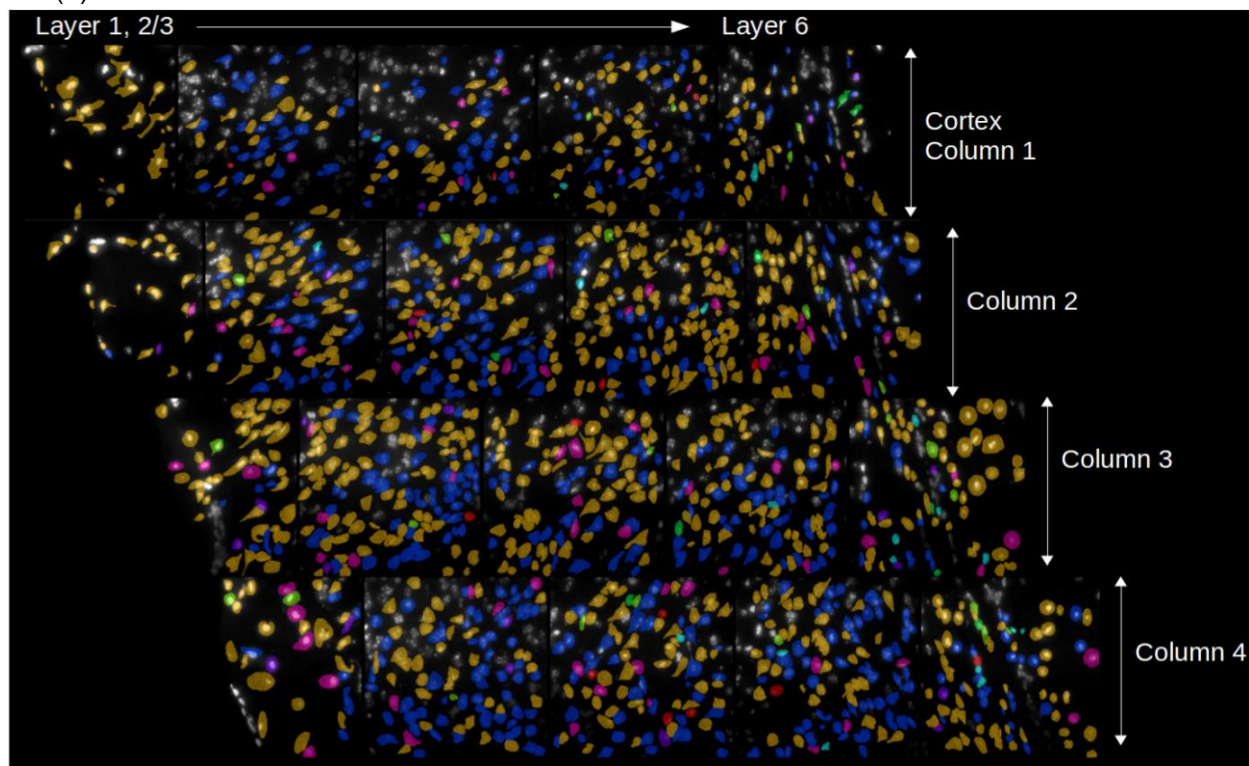
(b) Cortex column 1
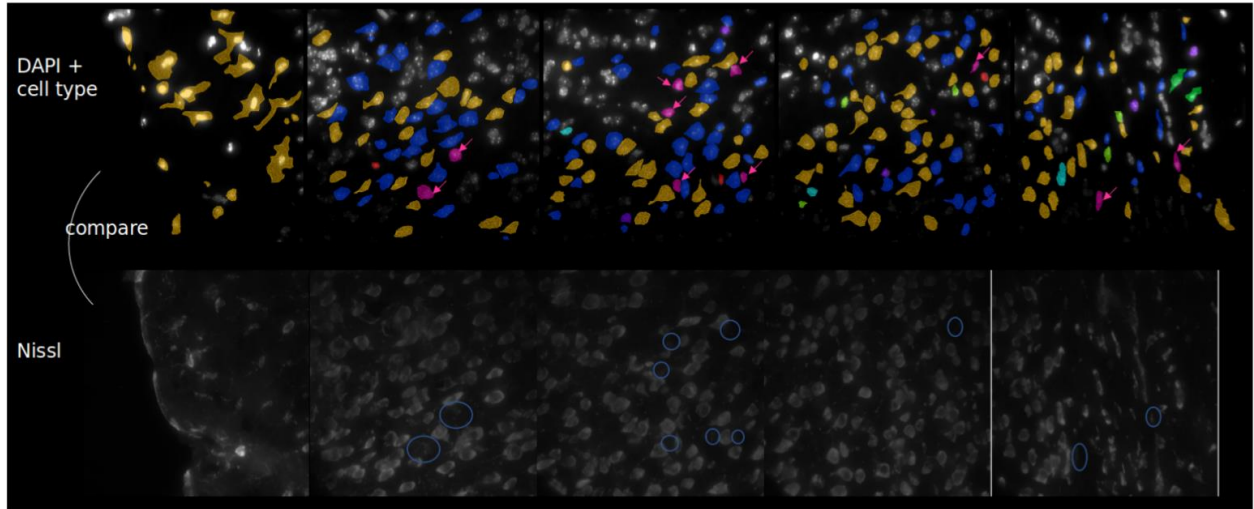
(c) Cortex column 2

(d) Cortex column 3

(e) Cortex column 4

DAPI + cell type

compare

Nissl

Mathematical details of the Hidden Markov random field model

We used the expectation-maximization (EM) algorithm for the estimation of parameters[1,2]. Namely, for fixed number of domain states k (set to be 9 in this paper), and smoothness parameter β (set to be 9.0 in this paper), the Gaussian mixture model parameters, mean μ_j and covariance matrix Σ_j, where $j=1...k$, were estimated using the following iterative procedure.

Initializations:
The domain configuration was initialized using the k-means clustering result. The parameter values for $\mu_{j\,(0)}$ and $\Sigma_{j\,(0)}$ for $j=1...k$ were estimated based on this initial configuration.

E-step, at m-th iteration:
For $j=1...k$:
Let $\phi_{j\,(m)} = \{\mu_{j\,(m)}, \Sigma_{j\,(m)}\}$
For node i:
Let $c_{i(m)}$ be the class (or label) of node i and $c_{N_i(m)}$ be the class of neighboring nodes at m-th iteration.
Let $\gamma_{i,j\,(m)}$ be the estimate at m-th iteration of the probability that cell i was derived from class j.

$$\gamma_{i,j\,(m)} = P(j|\phi_{j(m)}, c_{N_i(m)}) / \Sigma_{l=1...k}P(l|\phi_{l\,(m)}, c_{N_i(m)}) \qquad \text{Eq (1)}$$

Where $P(j|\phi_{j\,(m)}, c_{N_i(m)}) = P(j|x_i, c_{N_i(m)}) = P(x_i|j, \phi_{j(m)})P(j|c_{N_i(m)})$. Here $P(x_i|j, \phi_{j(m)})$ is the Gaussian density. $P(j|c_{N_i(m)}) = exp(-\beta\Sigma_{t\in N_i}U(c_{t(m)}, c_{i(m)}))$ is the neighborhood effect factor where the potential function $U(c_{t(m)}, c_{i(m)}) = -1, if\ c_{t(m)} = c_{i(m)};\ and\ 0\ otherwise$. The denominator in Eq(1) is the normalizing constant.

The Gaussian density is defined as: $P(x_i|j, \phi_{j(m)}) = 1/\sqrt{(2\pi)^N det(\Sigma_{j(m)})}\ exp(-1/2\ (x_i - \mu_{j(m)})^T \Sigma_{j(m)}^{-1}\ (x_i - \mu_{j(m)}))$

M-step, at m-th iteration:
Now based on the current estimate $\gamma_{i,j\,(m)}$, we next assign a class label for i based on the maximum *a priori* (MAP) criterion[1,3]:

$$c_{i(m+1)} = argmax_j\{\gamma_{i,j(m)}\}$$

We then estimate the parameters at $(m + 1)$ iteration as follows.
Let $n_{j(m)} = \Sigma_{i=1...n}\gamma_{i,j\,(m)}$
For $j=1...k$:

$$\mu_{j\,(m+1)} = 1/n_{j(m)}\ \Sigma_{i=1...n}(\gamma_{i,j(m)}x_i)$$
$$\Sigma_{j(m+1)} = 1/n_{j\,(m)}\Sigma_{i=1...n}(\gamma_{i,j(m)}(x_i - \mu_{j\,(m+1)})(x_i - \mu_{j\,(m+1)})^T)$$

Repeat step E and step M until the following stopping criteria is met.

Stopping criteria:
Set tolerance threshold ϵ=1e-8.
Compute:

$$relerr_\mu = max_j(\|\mu_{j(m+1)} - \mu_{j(m)}\|) / (1 + max_j(\|\mu_{j(m+1)}\|, \|\mu_{j(m)}\|))$$
$$relerr_\Sigma = max_j(\|\Sigma_{j(m+1)} - \Sigma_{j(m)}\|) / (1 + max_j(\|\Sigma_{j(m+1)}\|, \|\Sigma_{j(m)}\|))$$

If $relerr_\mu < \epsilon$ and $relerr_\Sigma < \epsilon$, then stopping criteria is met, and the algorithm is stopped.

Once the parameter values are set, the domain states are determined as the set of $c_{i(m)}$.

Consideration of ill-conditioning covariance matrix

Occasionally, the covariance matrix Σ may be ill-conditioned causing numerical instability. This case can occur if a large portion of data points are highly co-linear. To improve numerical stability, we add a regularizing term along the diagonal of the covariance matrix Σ, so that it becomes $\Sigma + Id$. The damping coefficient *d* is initially set to be equal to a small value (1e-5). If the regularized matrix remains ill-conditioned, that is, $det(\Sigma + Id) <$ c where *c* is a small number (1e-30), then we continue to increase the value of *d* (by a factor of 1.05 each time) until $det(\Sigma + Id) > c$ is satisfied. Otherwise, if initial check already yields $det(\Sigma + Id) > c$, we decreased *d* (by a factor 1.05 at a time) until *d* is the minimum value that satisfies $det(\Sigma + Id) > c$.

During the EM-procedure, we also follow a fixed node update order. This order is determined by the chromatic number of the neighborhood graph, which is the minimum number of colors to color the nodes of the graph such that two adjacent nodes have different colors[4]. Nodes labeled with the same color are updated by EM first, then we move to nodes labeled with another color, and so forth. This step has been done previously in mritc package[4], to increase speed (as independent regions of the graph can be updated at the same time) and to improve consistency (because independence between adjacent nodes helps reduce interference).

1. Zhang, Y., Brady, M. & Smith, S. Segmentation of Brain MR Images Through a Hidden Markov Random Field Model and the Expectation-Maximization Algorithm. *Ieee* **20,** 45–57 (2001).
2. Li, S. Z. Modeling image analysis problems using Markov random fields. *Stoch. Process. Model. Simul.* 473 (2003).
3. Wang, Q. HMRF-EM-image: Implementation of the Hidden Markov Random Field Model and its Expectation-Maximization Algorithm. *arXiv Prepr.* (2012).
4. Feng, D., Tierney, L. & Magnotta, V. MRI Tissue Classification Using High-Resolution Bayesian Hidden Markov Normal Mixture Models. *J. Am. Stat. Assoc.* **107,** 102–119 (2012).