Applications of statistical physics concepts to quantifying neuron location, size, and shape by computer

Andrew Inglis

collaborators:

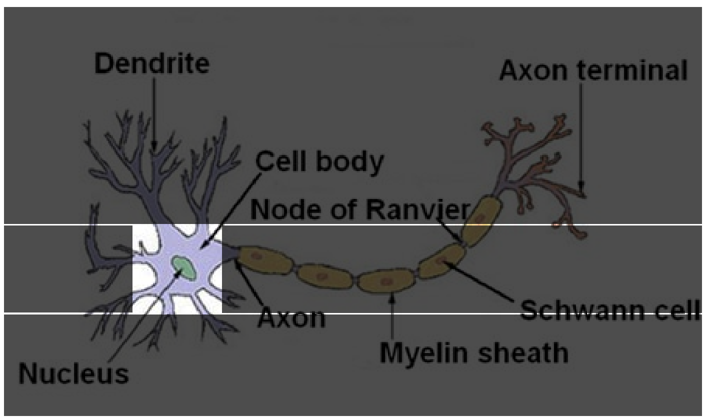
Daniel Roe, Brigita Urbanc, Luis Cruz, H.E.Stanley, Douglas Rosene

Question:

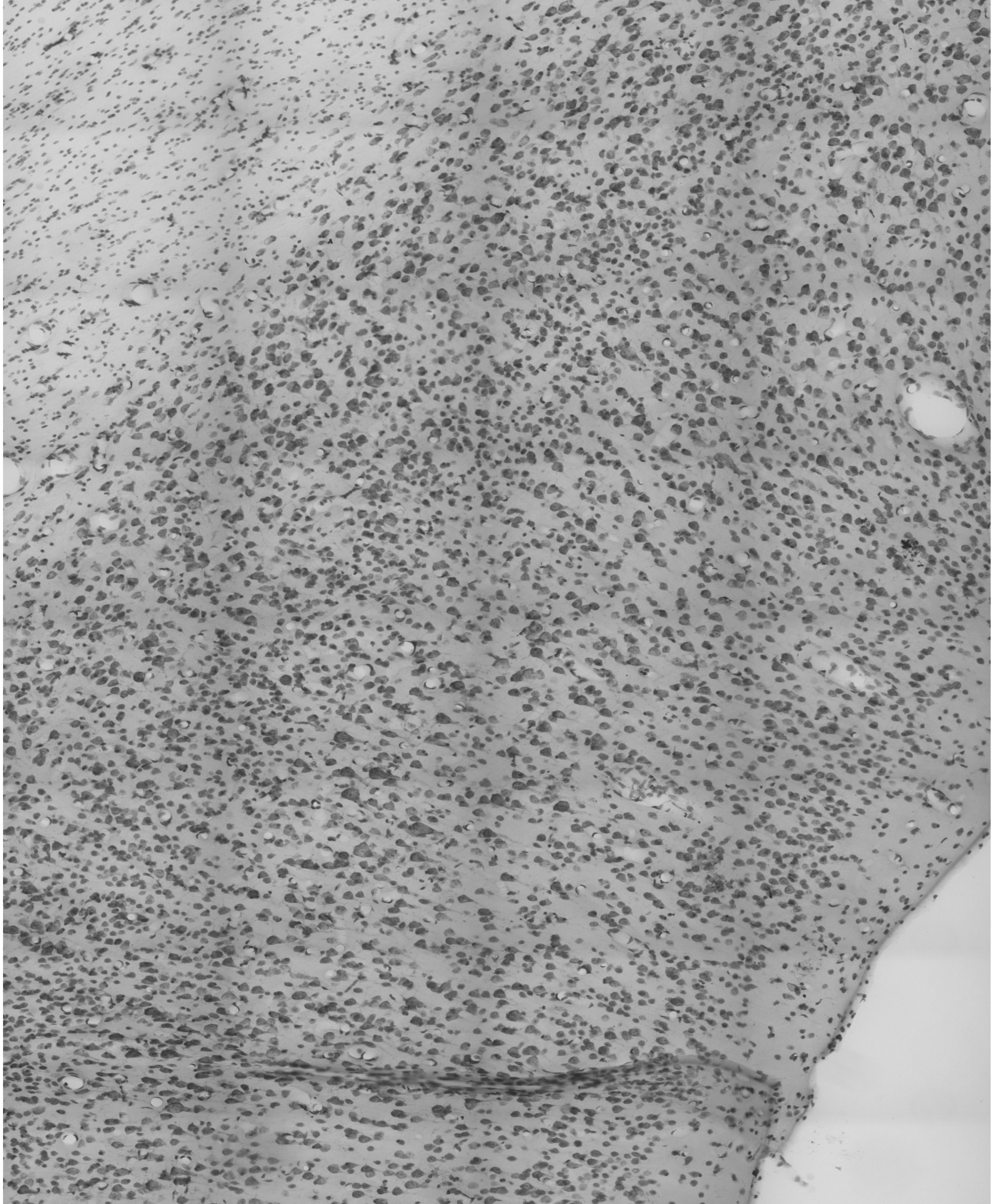
How can we train a computer to automatically identify neurons and determine coordinates, size, and shape?

Motivation:

- We want to improve the knowledge of how the brain works. Specifically, how neurons function together.
- We want to find patterns in neuron spatial organization.
- Analysis requires large datasets of individual neuron properties (10^3 - 10^{10} neurons).
- Traditional sampling methods (ie: stereological) only measure average quantities.

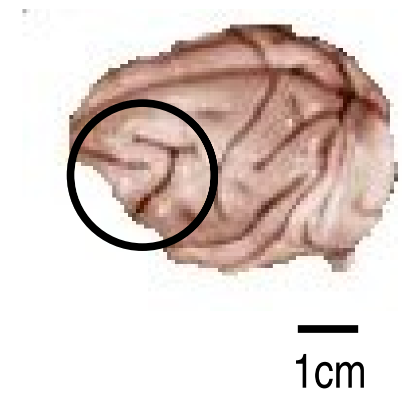


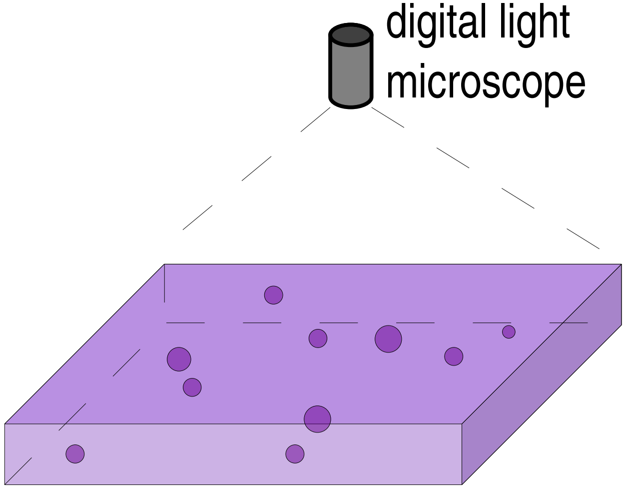
Structure of a typical neuron



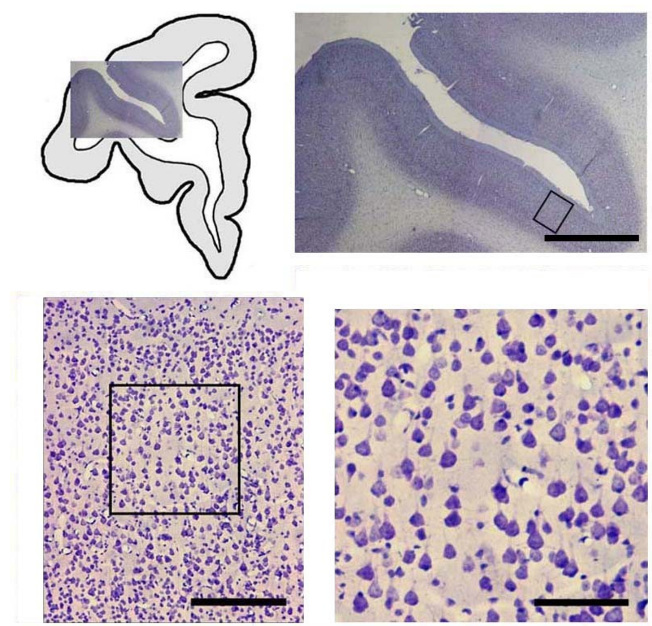
Input

- slice brain tissue (30μm sections) from *postmortem* samples.
- apply staining for neuron cell bodies (ie: Nissl staining).
- digital images obtained by light microscope.
- convert to gray scale (value varies from 0-255 per pixel)

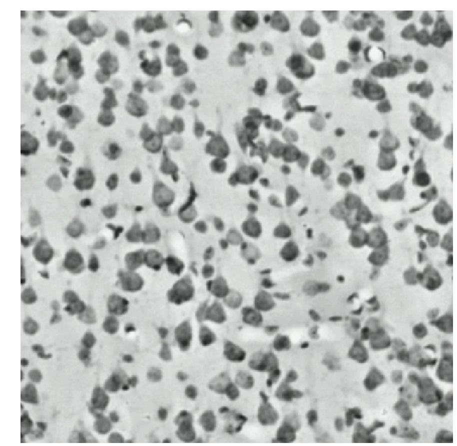




tissue sample: 500 x 500 x 30 μm
typical neuron diameter: 10μm



(image courtesy of Daniel Roe)

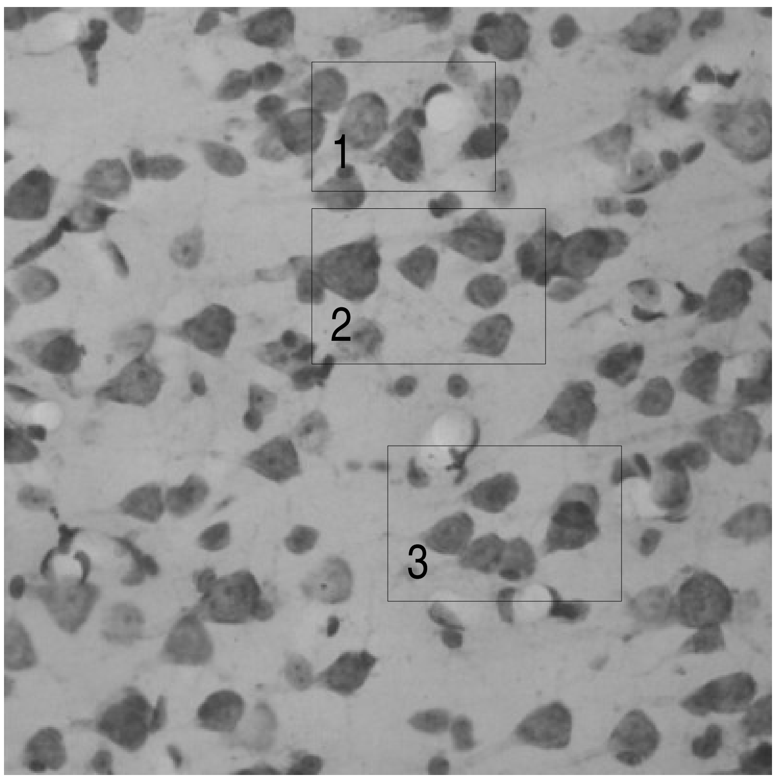


gray scale image
2^{18} pixels (512x512)

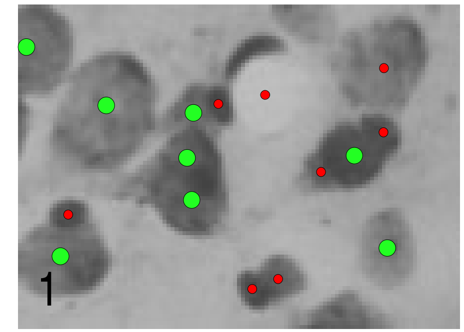
from image, obtain x,y location, size, and shape information for each neuron

Challenges to recognizing and locating neuron bodies

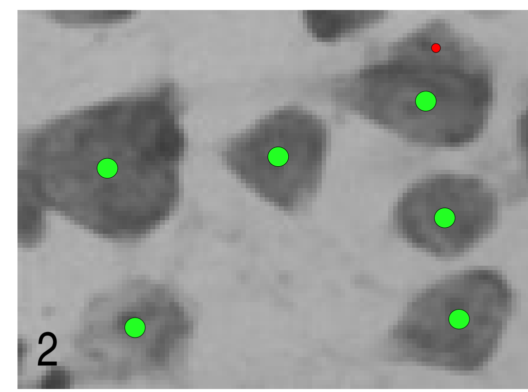
neuron · not neuron



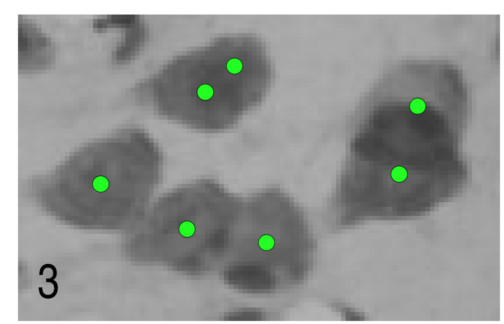
ignoring other cells and artifacts



neuron diversity

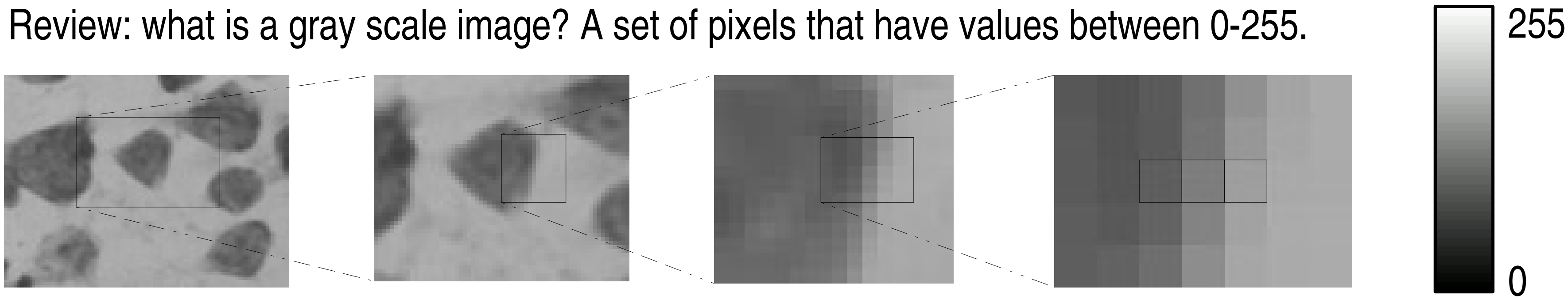


neuron overlapping



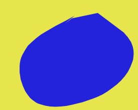
Finding neurons using clusters

Review: what is a gray scale image? A set of pixels that have values between 0-255.

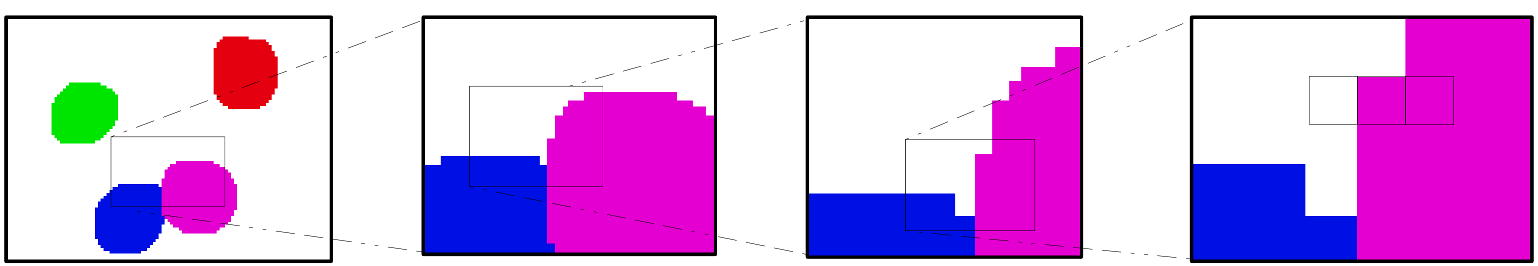


note: pixels are individual features that are not *a priori* connected to each other.

Color Cluster: a set of pixels that are connected by color.



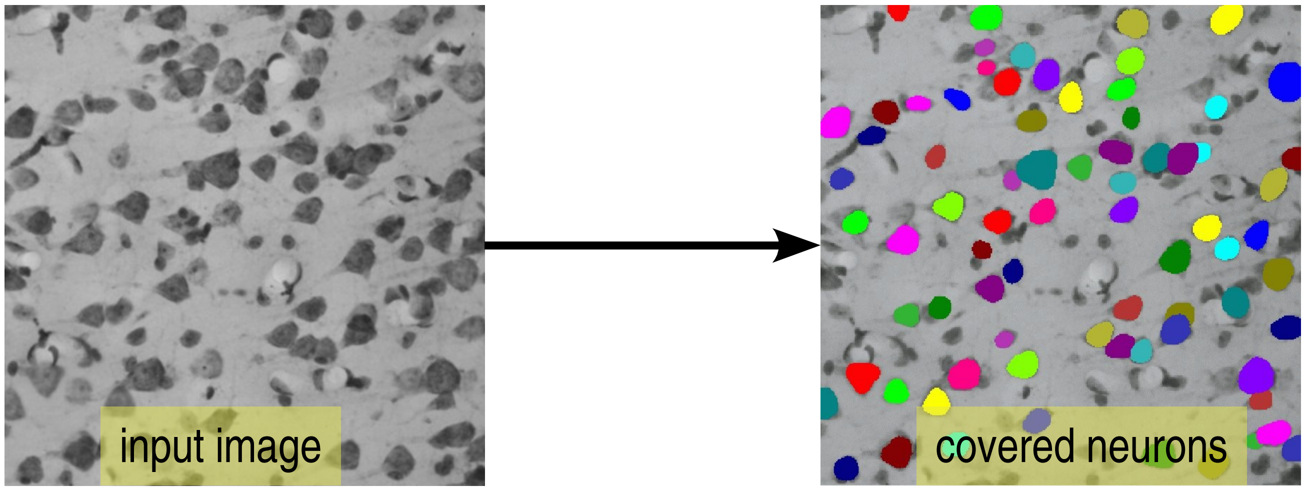
color cluster example



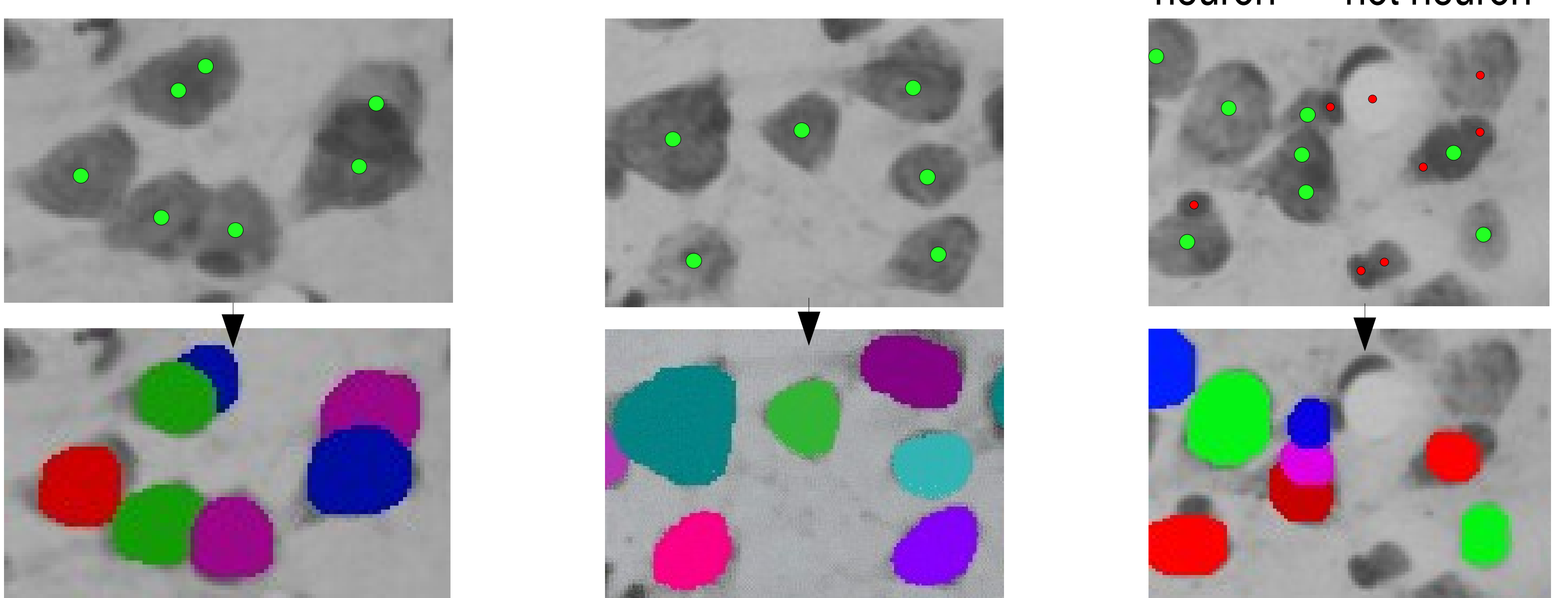
Different colors distinguish different clusters – like countries on an atlas

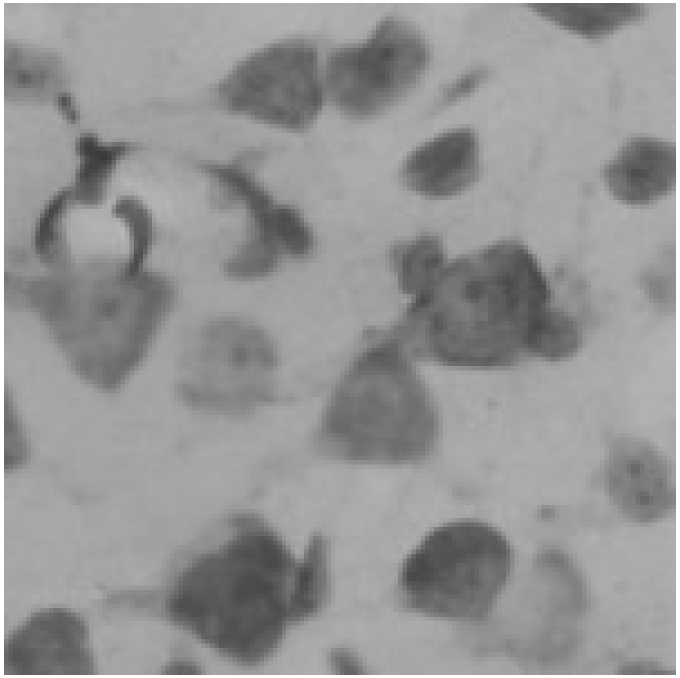
Goal: create clusters that overlap the neuron bodies.

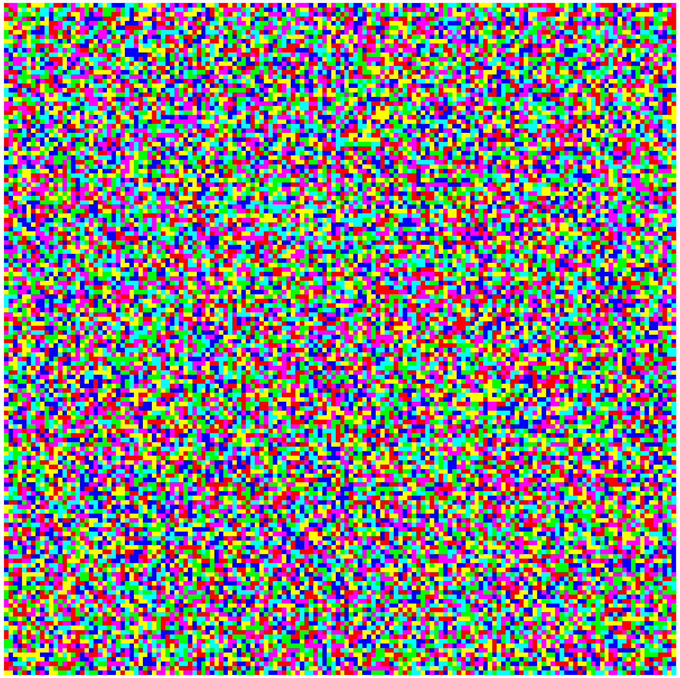
Ideal results for a cluster-finding algorithm:

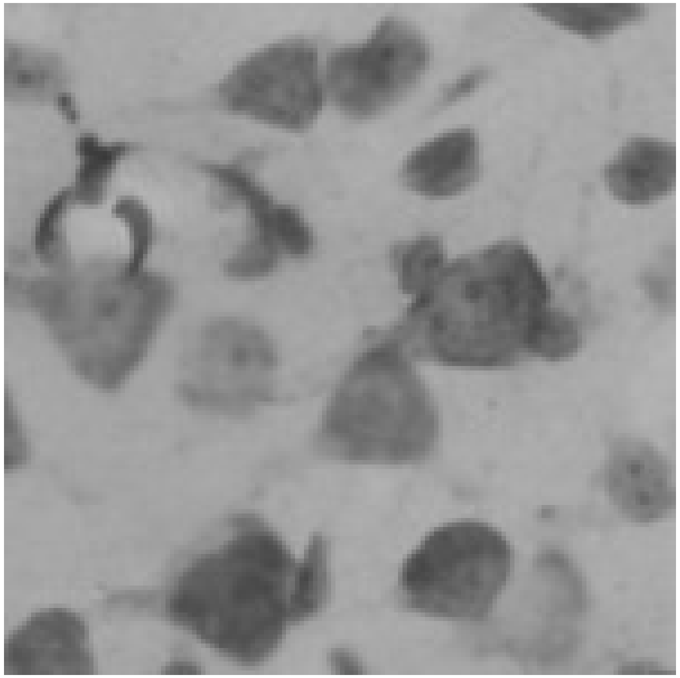
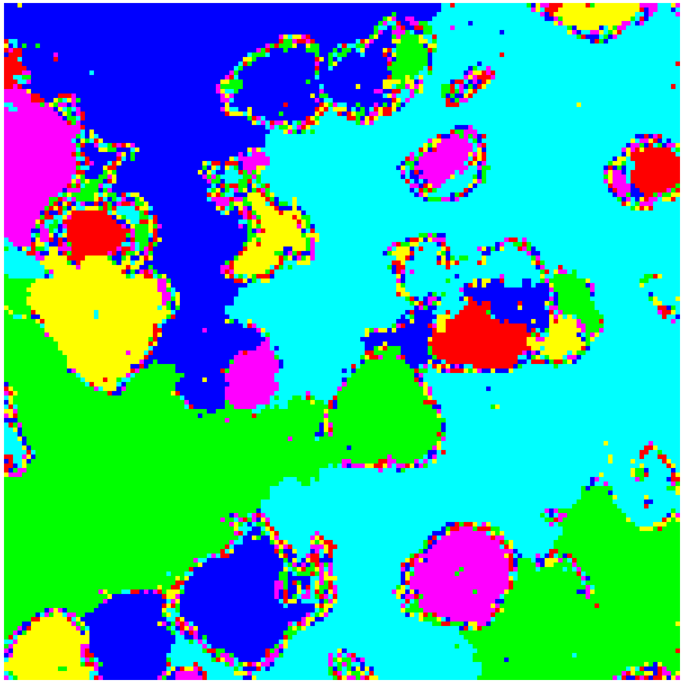


in more detail...





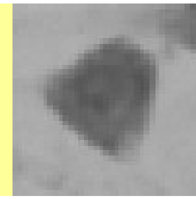


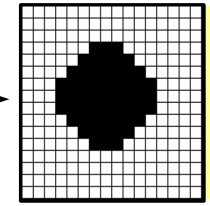
Using Potts segmentation to cluster images

...start with a simple example

1600 pixels (40x40)
gray scale values range
from 0 *to* 255



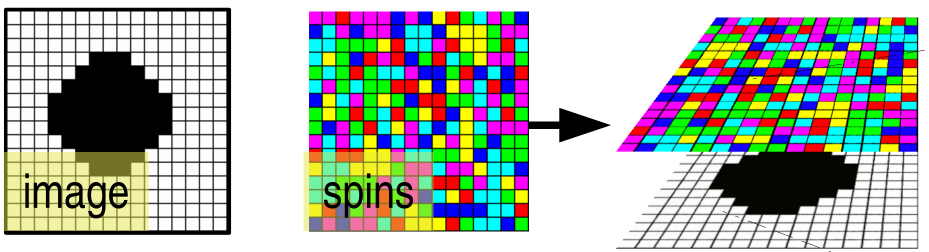
actual neuron

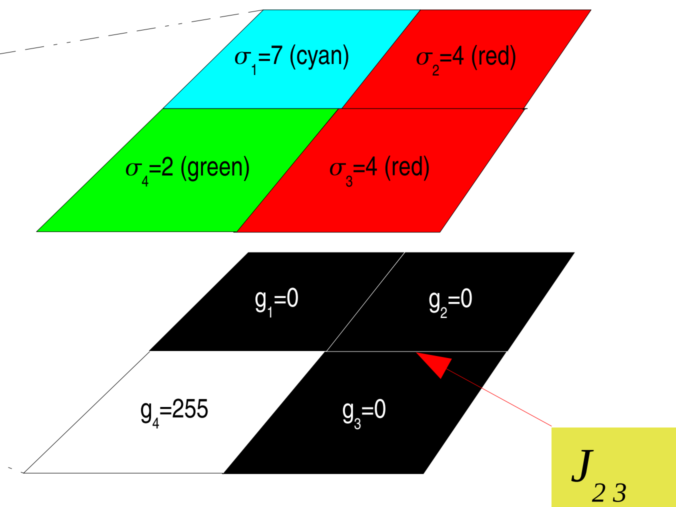


"toy" neuron

256 pixels (18x18).
gray scale values
either 0 *or* 255

- Overlay a lattice of random spin states over the image





- for each image pixel i, define gray scale value g_i
- for each nearest neighbors i and j, define strength $J_{ij} = 1 - \dfrac{|g_i - g_j|}{\theta\langle g_i - g_j\rangle}$
- for each spin site i, define "color" σ_i
- define Hamiltonian of system $H = -\sum_{\langle i,j\rangle} J_{ij}\,\delta_{\sigma_i \sigma_j}$

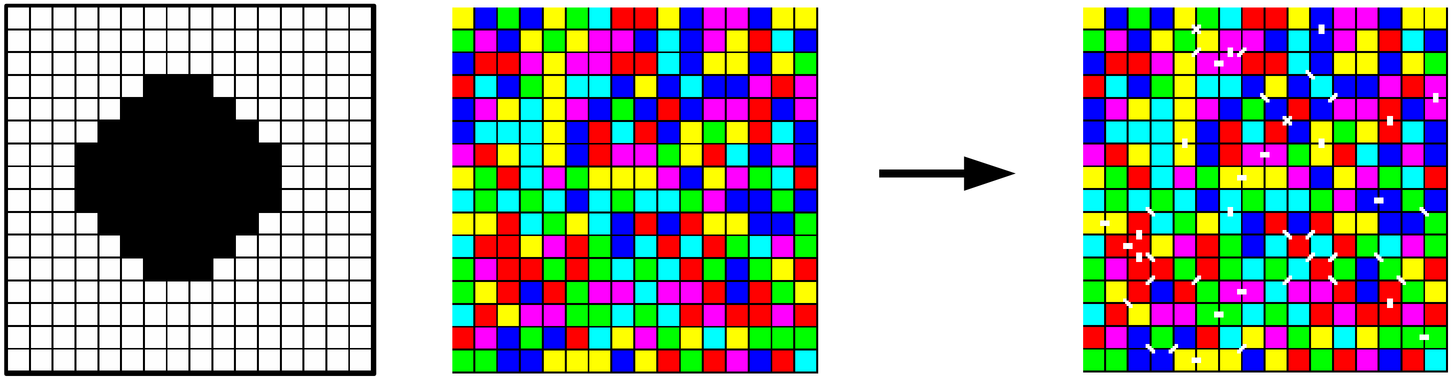
θ = threshold constant

$$\delta_{\sigma_i \sigma_j} = \begin{cases} 1 & \sigma_i = \sigma_j \\ 0 & \sigma_i \neq \sigma_j \end{cases}$$

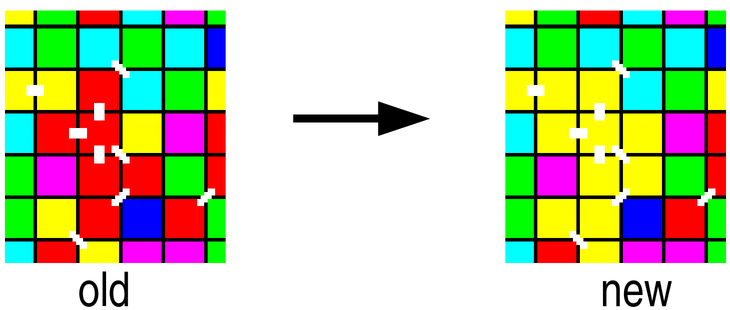
Ferber and Worgotter. 1998

Using Potts segmentation to cluster images

- freeze bonds between color sites with probability $(1-e^{-\beta J_{ij}})\delta_{\sigma_i \sigma_j}$ to form **frozen bond clusters**



- "flip" the **frozen bond cluster** to a new color σ (example: red to yellow).



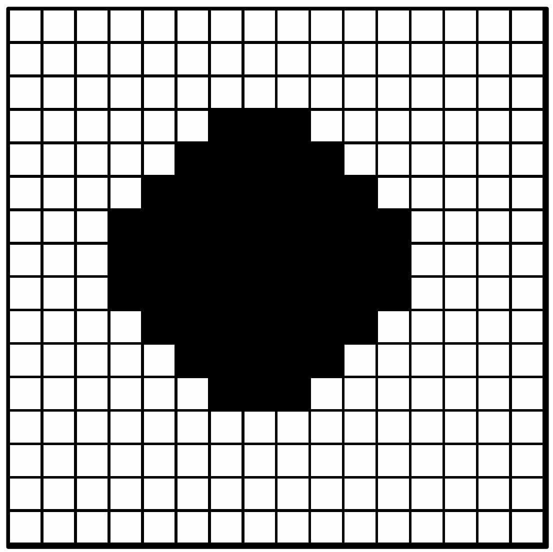
- calculate the change in energy H due to the cluster color "flip" $\Delta H = H_{new} - H_{old}$.
- if $\Delta H < 0$, keep new color with probability 1.
- If $\Delta H > 0$, keep new color with probability $e^{-\beta \Delta H}$.

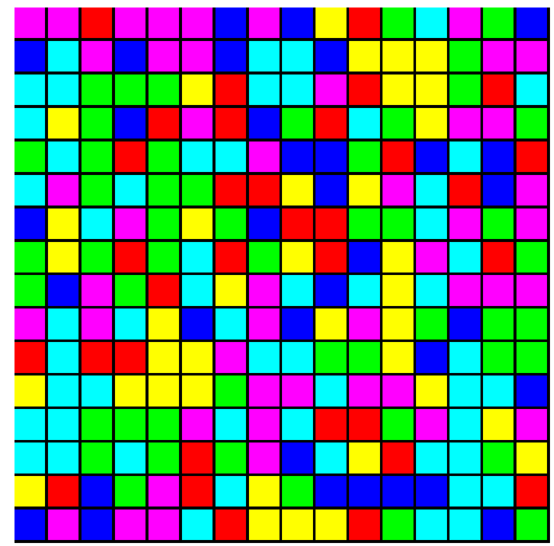
Hence, neighboring spins corresponding to similar pixels tend to align

Ferber and Worgotter. 1998

Using Potts segmentation to cluster images

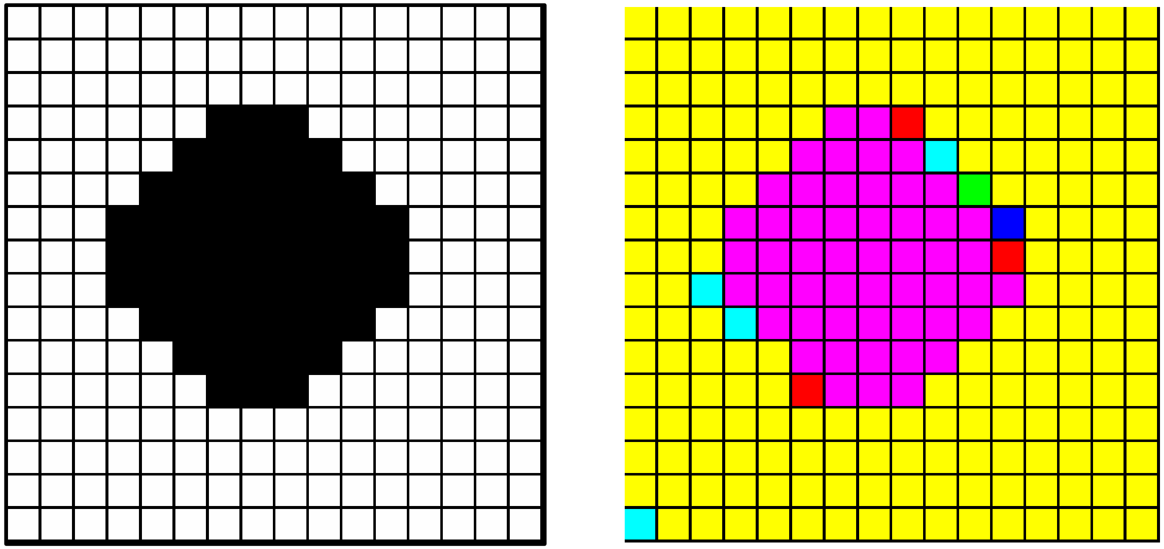
- 50 iterations. Spin States = 6. Start from random spin ("color") configuration.





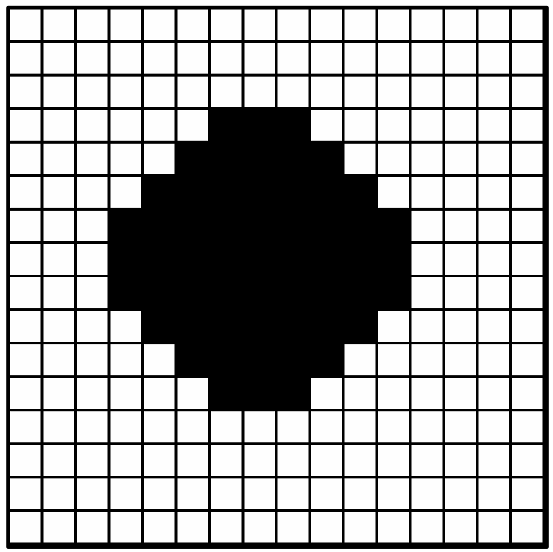
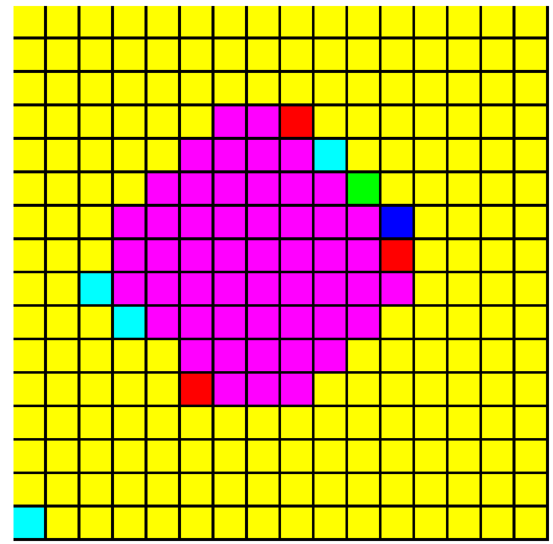
Using Potts segmentation to cluster images

- 50 iterations. Spin States = 6. Start from random spin (“color”) configuration.

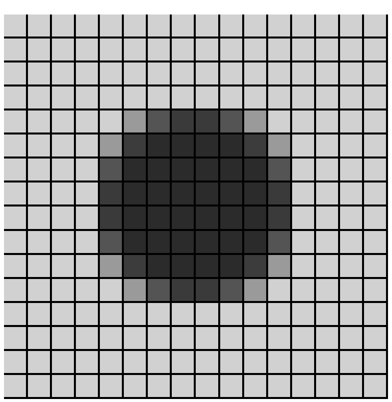
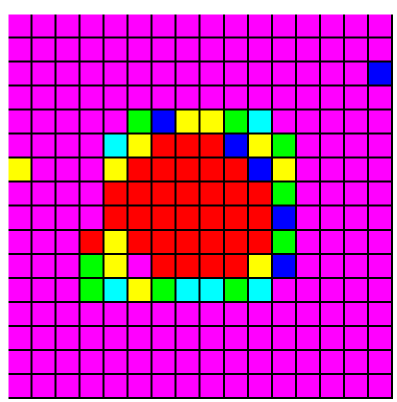
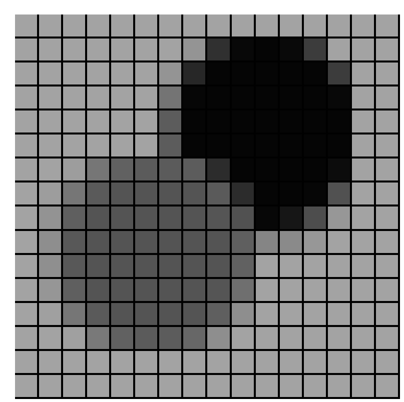
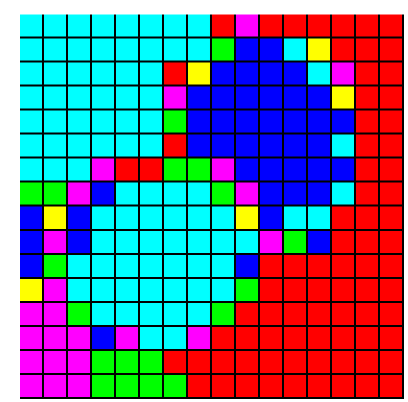


Using Potts segmentation to cluster images

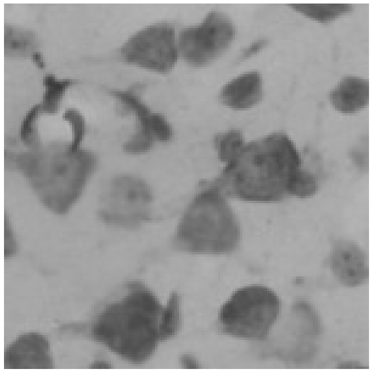
- 50 iterations. Spin States = 6. Start from random spin ("color") configuration.

- other examples...

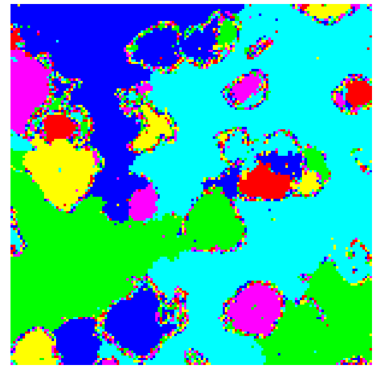





Neuron recognition using Potts clustering



$150 \times 150 = 2^{15}$ pixels

$\theta = 1$
$\beta = 1$



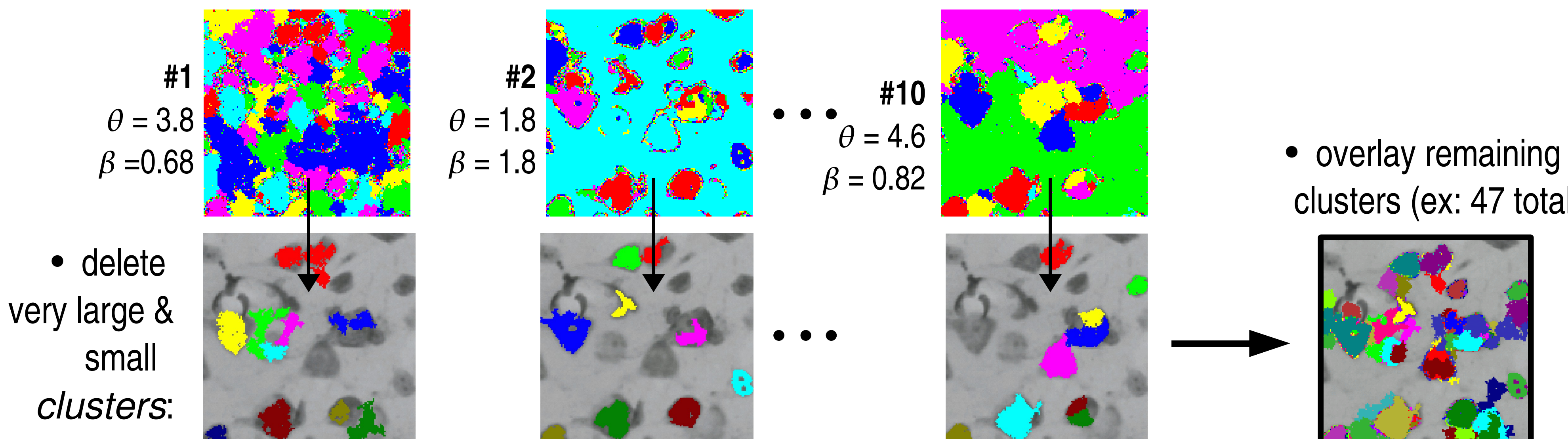
final state

$$J_{ij} = 1 - \frac{|g_i - g_j|}{\theta \langle g_i - g_j \rangle}$$

$$P_{freeze} = (1 - e^{-\beta J_{ij}})\, \delta_{\sigma_i \sigma_j}$$

$$P_{flip} = e^{-\beta \Delta H}$$

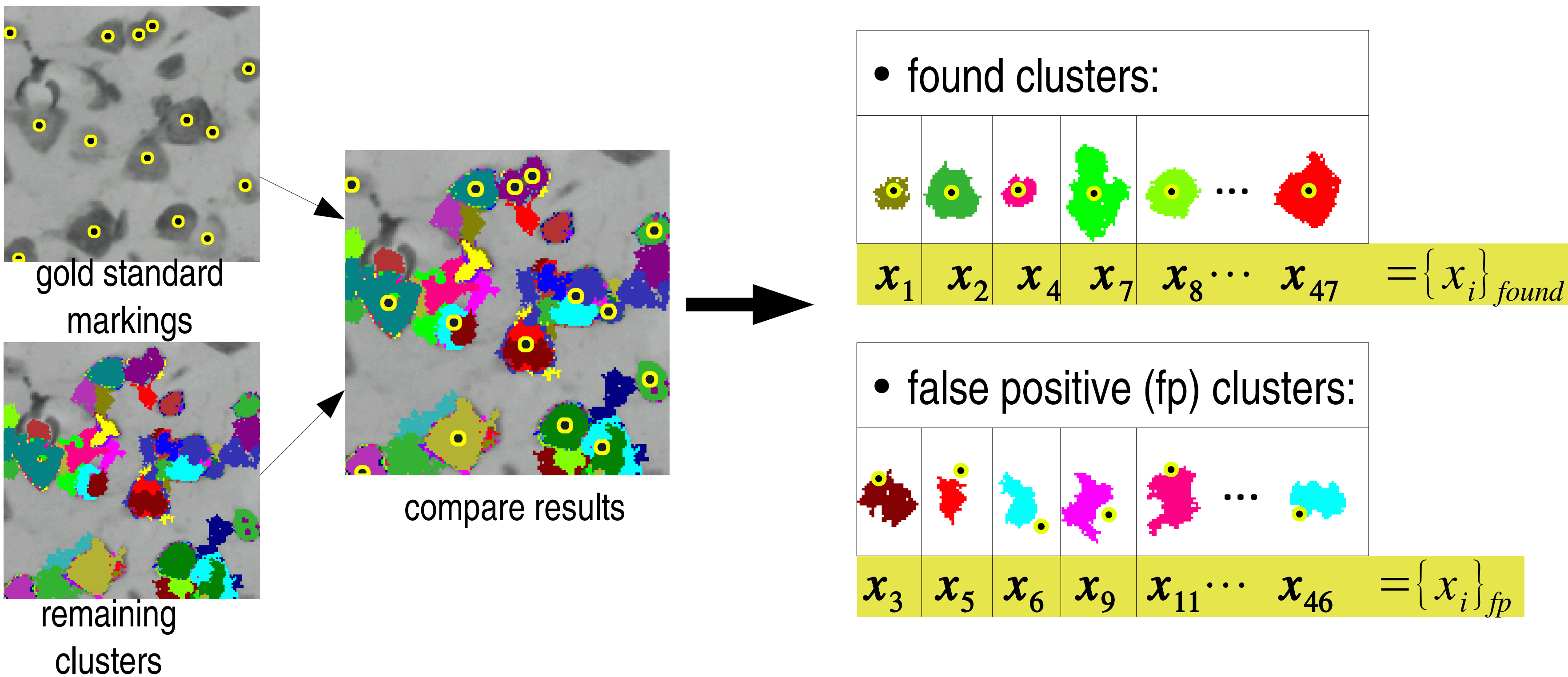
Parallelization: create final states of different θ,β parameters (ex: 10 choices)

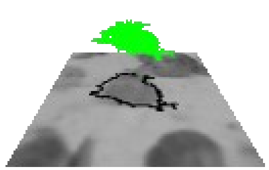


Peng et al. 2003

Cluster selection by computer training

- Make image a “gold standard” : a tissue sample marked by a neuroanatomist





example cluster

$$x_i = \begin{pmatrix} \text{area} \\ \text{avg. grayscale} \\ \text{grayscale variance} \\ \text{grayscale 'gyration'} \\ \text{gyration} \\ \text{edge length vs. area} \end{pmatrix}$$

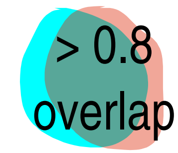
- use $\{x_i\}_{found}$ to find probability distribution $P_{found}(x)$
- use $\{x_i\}_{fp}$ to find probability distribution $P_{fp}(x)$

<u>Gaussian Mixture Models</u>. (Dempster et al. 1977)

Finding neurons for any image (...that looks like the “gold standard”)

1. overlay final states from θ/β runs.
2. if cluster i is very small or very large, then DELETE.
3. if $P_{fp}(x_i) > P_{found}(x_i)$ then DELETE cluster i.
4. if x_i and x_j ($i \neq j$) overlap by > 0.8, and $P_{found}(x_j) > P_{found}(x_i)$, then DELETE cluster i.

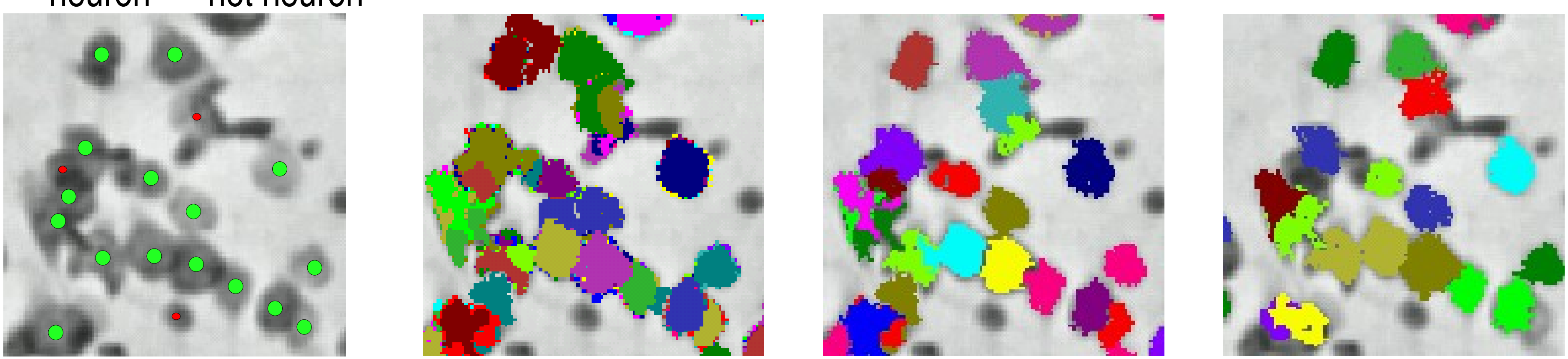
<u>remaining clusters represent neurons.</u>



ex: two clusters compete for the same feature

an example...

neuron • not neuron

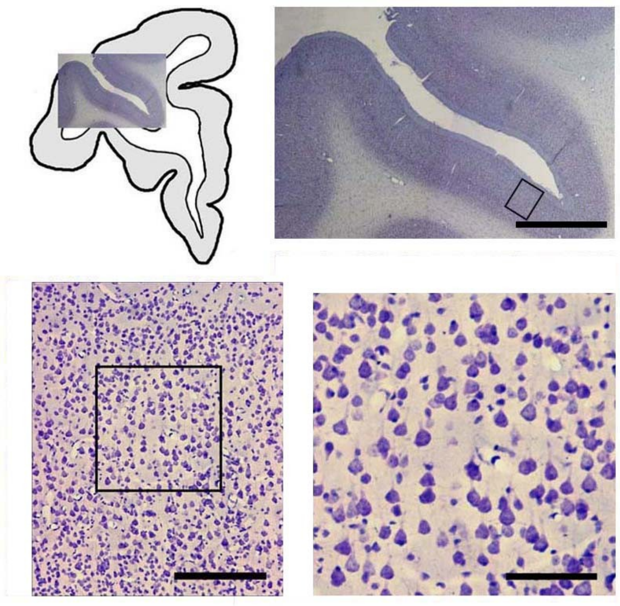


initial image | after step **2.** | after step **3.** | after step **4.** **(FIN)**

Method validation

<u>Data Set</u>

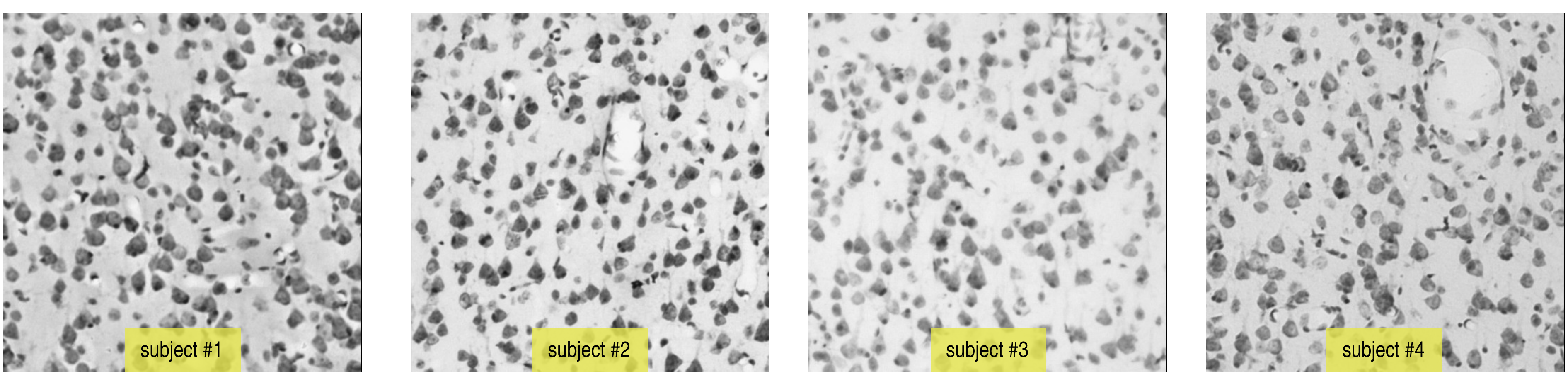
- prefontal cortex (area 46, layer 3) from Rhesus Monkey. 15 subjects of varying age, 8 pictures each = 120 total images.
- each image 2^9 x 2^9 (512x512) pixels with resolution 1 μm per pixel length, ~150 neurons per image.
- previously marked using semiautomated methods. Used for neuron-neuron correlation density map analysis (*Cruz. et al 2004.)*
- Randomly select 7 out of 15 subjects for analysis



scale b=2.5mm, c=250 microns, d=100 microns

image courtesy of Daniel Roe

4 (out of 7) examples of variability of images type between subjects:



Method validation

For each subject:

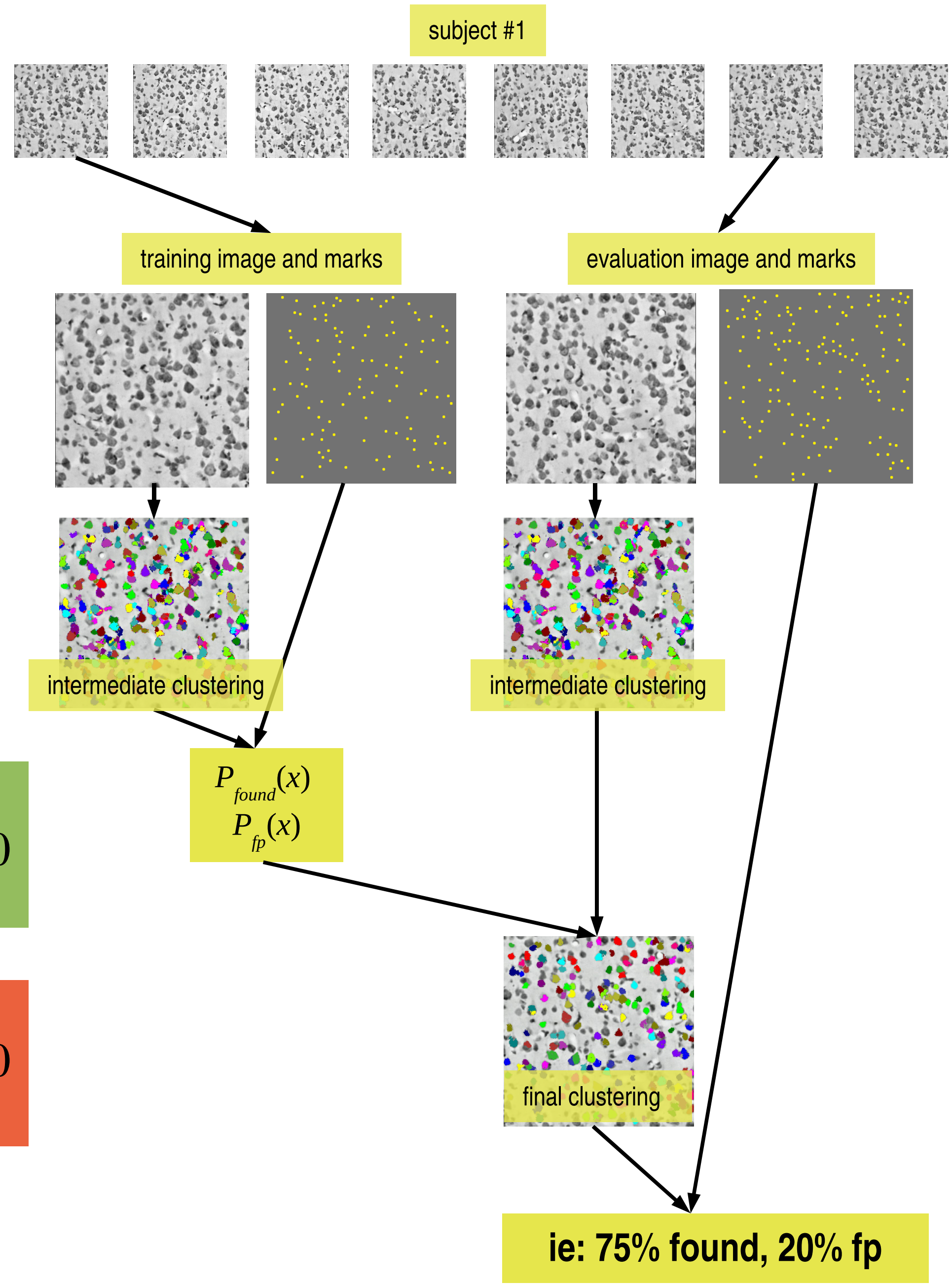
- randomly select *training* image and *evaluation* image from original 8 pictures.
- Neuroanatomist marks both images.
- use *training* image and marks to create $P_{found}(x_j)$ and $P_{fp}(x_i)$.
- analyze evaluation image.
- Test *evaluation* image cluster results compared to marks:

found %:
(efficiency %)

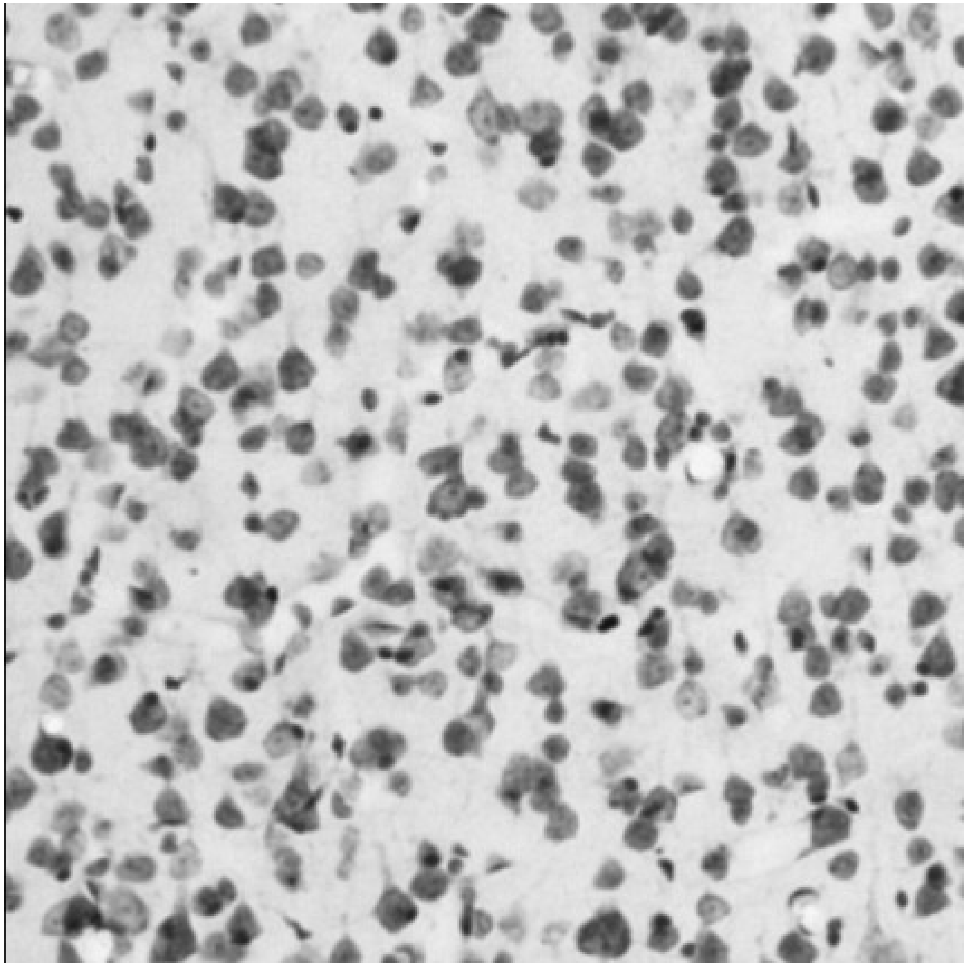
$$\frac{no.\ of\ neurons\ found}{no.\ of\ gold\ standard\ neurons} \cdot 100$$

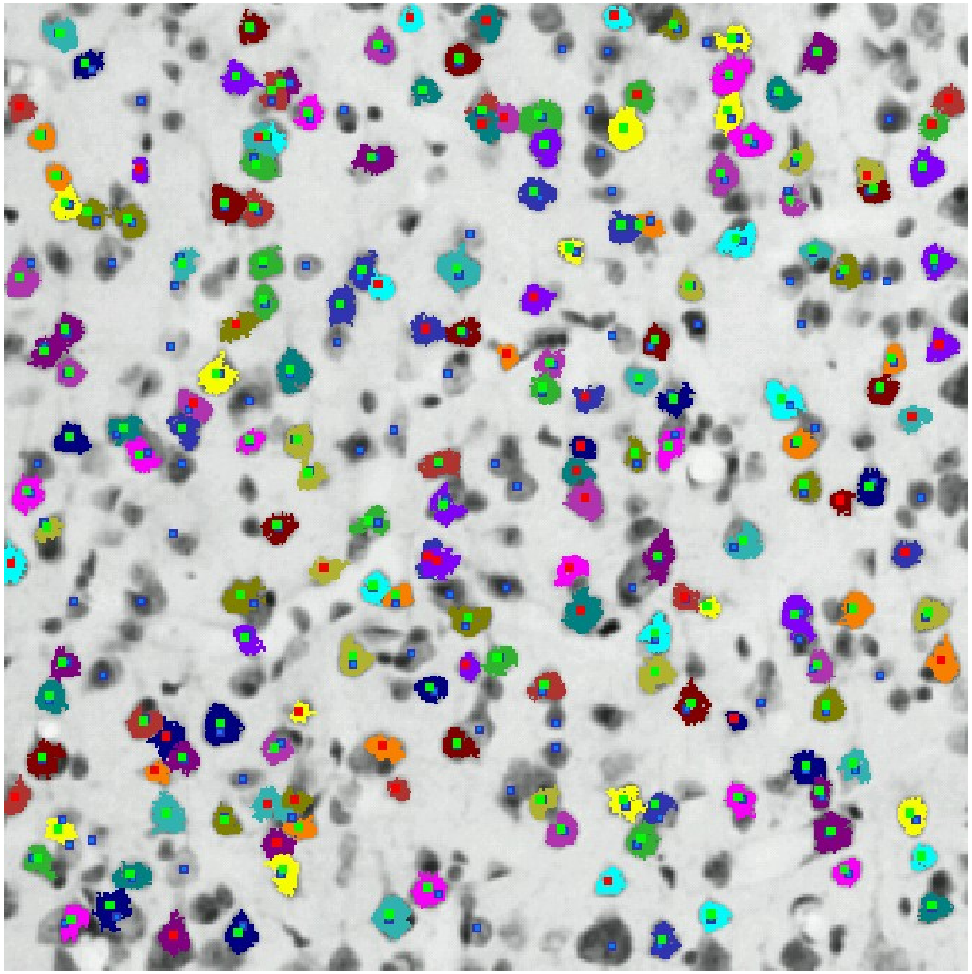
fp %:
(100- purity %)

$$\frac{no.\ of\ false\ positives}{no.\ of\ gold\ standard\ neurons} \cdot 100$$



Method validation



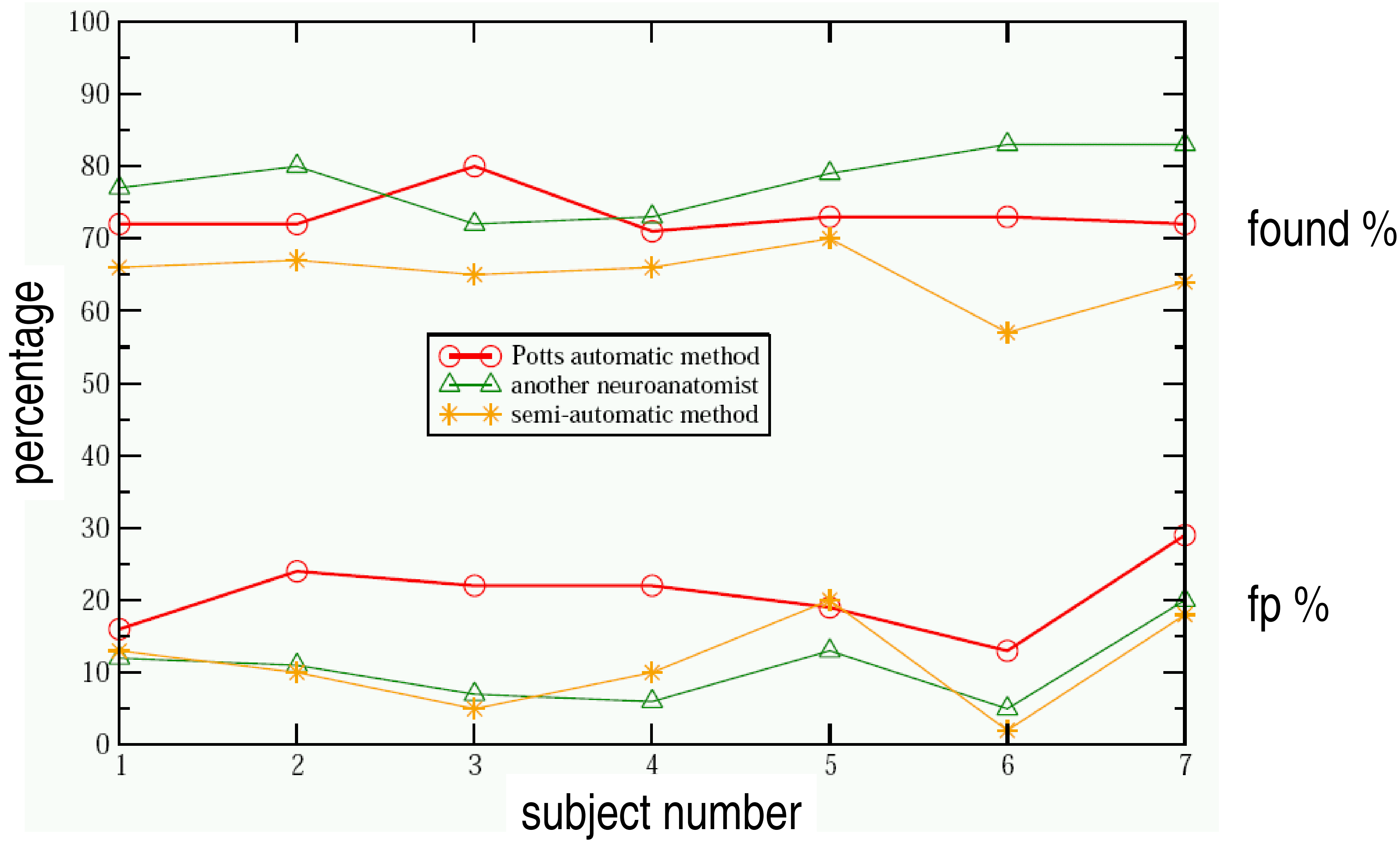


manual marks by neuroanatomist

neuron found by the computer

false positive found by the computer

Method validation



Potts: found : 73±3 % fp: 21±5%

another neuroanatomist: found : 78±4 % fp: 11±6%

semi-automatic method: found : 65±4% fp: 11±6%

Conclusions so far

A combination of using:

- Potts segmentation
- parallelization, and
- a trained network

allows for determination of individual neuron location in diverse tissue samples with the desired accuracy.

Next Steps

- Replace Potts clustering with quicker more efficient clustering method.
- Apply the method to correlative measurements.
- Apply the method to larger samples not possible for manual marking.
- Extend the method to measure neuron size, shape, and location within a 3-dimensional framework .
- Explore other trained network models in order to reduce recognition error.

Applications of statistical physics concepts to quantifying neuron location, size, and shape by computer

Andrew Inglis

collaborators:

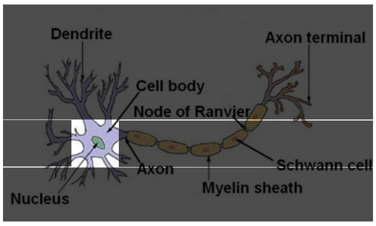
Daniel Roe, Brigita Urbanc, Luis Cruz, H.E.Stanley, Douglas Rosene

Question:

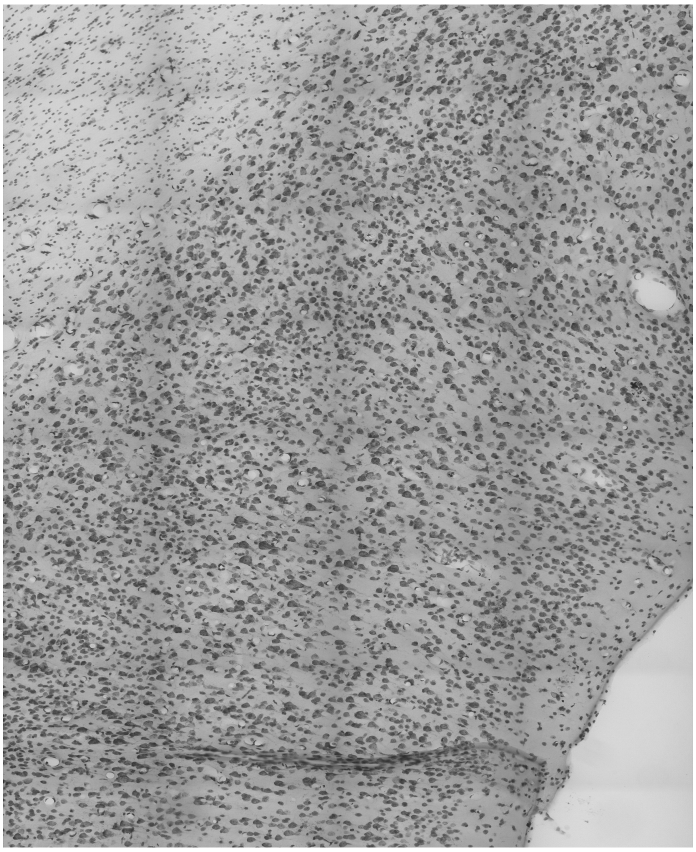
How can we train a computer to automatically identify neurons and determine coordinates, size, and shape?

Motivation:

- We want to improve the knowledge of how the brain works. Specifically, how neurons function together.
- We want to find patterns in neuron spatial organization.
- Analysis requires large datasets of individual neuron properties (10^3 - 10^{10} neurons).
- Traditional sampling methods (ie: stereological) only measure average quantities.

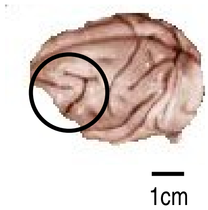


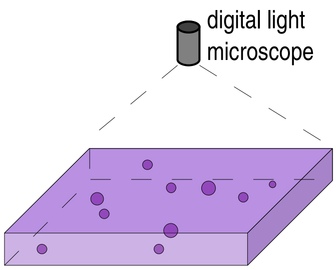
Structure of a typical neuron



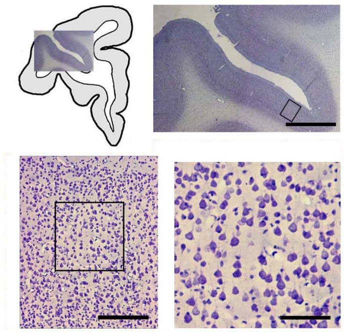
Input

- slice brain tissue (30μm sections) from *postmortem* samples.
- apply staining for neuron cell bodies (ie: Nissl staining).
- digital images obtained by light microscope.
- convert to gray scale (value varies from 0-255 per pixel)

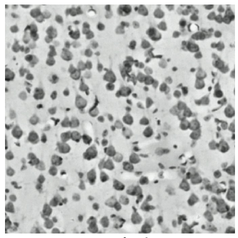




tissue sample: 500 x 500 x 30 μm
typical neuron diameter: 10μm



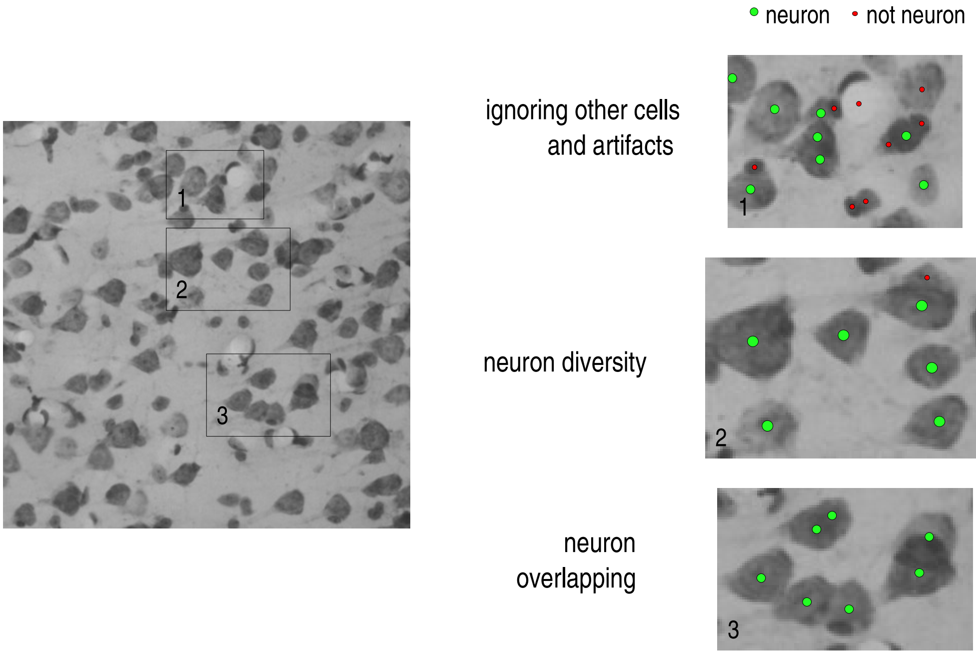
(image courtesy of Daniel Roe)



gray scale image
2^{18} pixels (512x512)

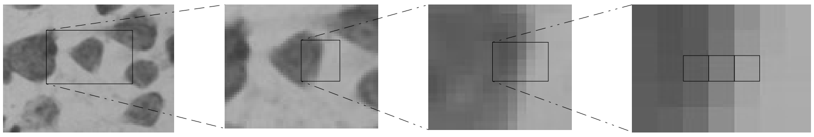
from image, obtain x,y location, size, and shape information for each neuron

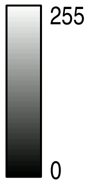
Challenges to recognizing and locating neuron bodies



Finding neurons using clusters

Review: what is a gray scale image? A set of pixels that have values between 0-255.

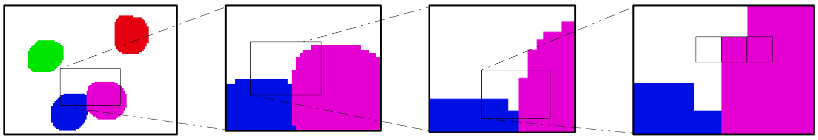




note: pixels are individual features that are not *a priori* connected to each other.

Color Cluster: a set of pixels that are connected by color.

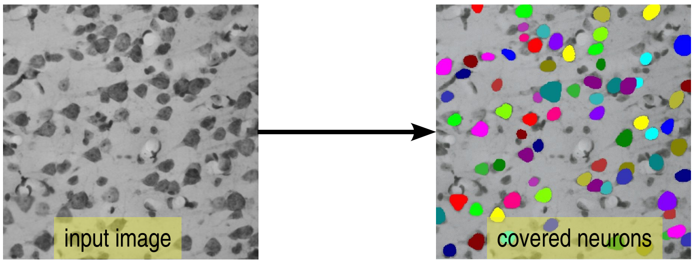
color cluster example



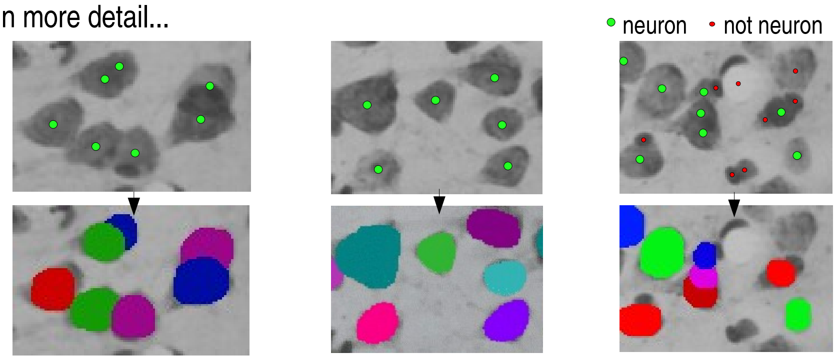
Different colors distinguish different clusters – like countries on an atlas

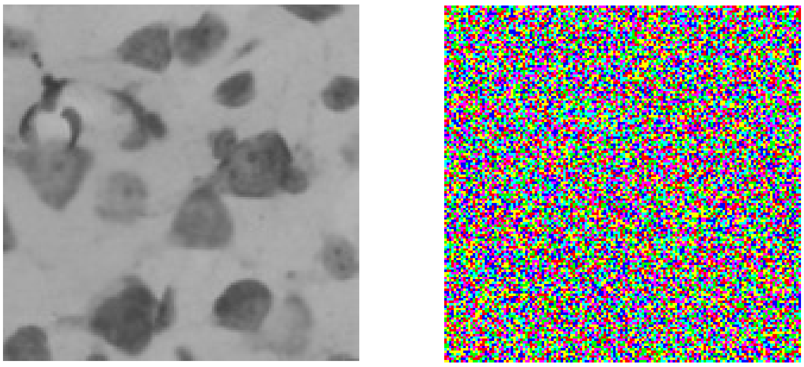
Goal: create clusters that overlap the neuron bodies.

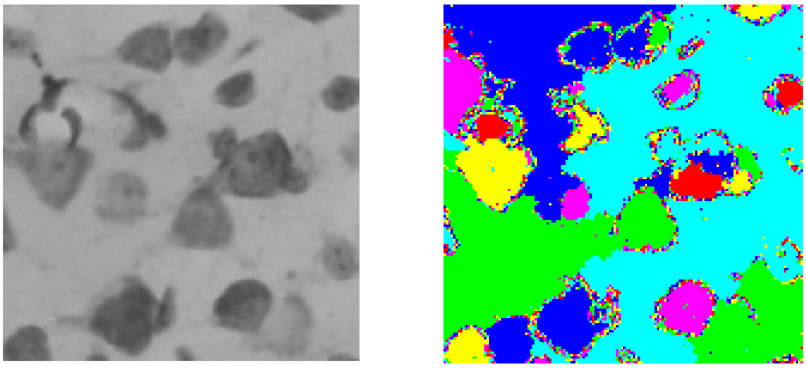
Ideal results for a cluster-finding algorithm:



in more detail...



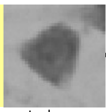




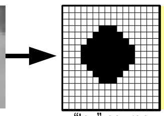
Using Potts segmentation to cluster images

...start with a simple example

1600 pixels (40x40)
gray scale values range
from 0 *to* 255



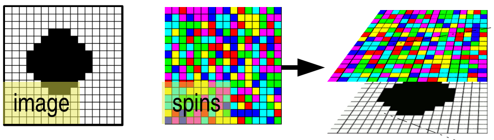
actual neuron

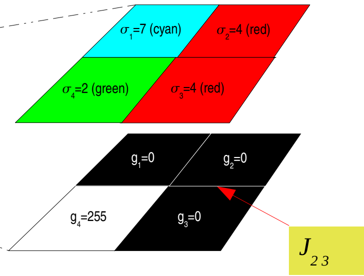


"toy" neuron

256 pixels (18x18).
gray scale values
either 0 *or* 255

- Overlay a lattice of random spin states over the image





- for each image pixel i, define gray scale value g_i
- for each nearest neighbors i and j, define strength $J_{ij} = 1 - \frac{|g_i - g_j|}{\theta \langle g_i - g_j \rangle}$
- for each spin site i, define "color" σ_i
- define Hamiltonian of system $H = -\sum_{\langle i,j \rangle} J_{ij} \delta_{\sigma_i \sigma_j}$

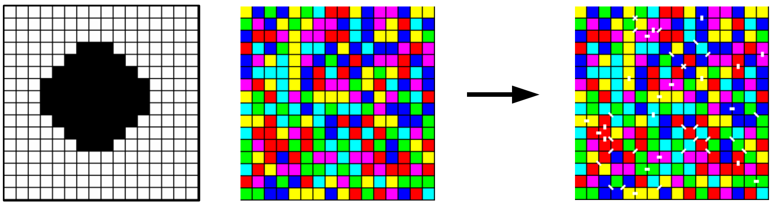
θ = threshold constant

$$\delta_{\sigma_i \sigma_j} = \begin{cases} 1 & \sigma_i = \sigma_j \\ 0 & \sigma_i \neq \sigma_j \end{cases}$$

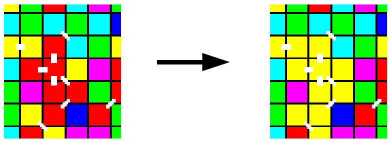
Ferber and Worgotter. 1998

Using Potts segmentation to cluster images

- freeze bonds between color sites with probability $(1-e^{-\beta J_{ij}})\,\delta_{\sigma_i \sigma_j}$ to form **frozen bond clusters**



- "flip" the **frozen bond cluster** to a new color σ (example: red to yellow).



old new

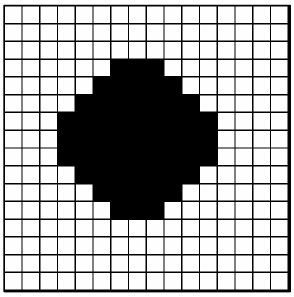
- calculate the change in energy H due to the cluster color "flip" $\Delta H = H_{new} - H_{old}$.
- if $\Delta H < 0$, keep new color with probability 1.
- If $\Delta H > 0$, keep new color with probability $e^{-\beta \Delta H}$.

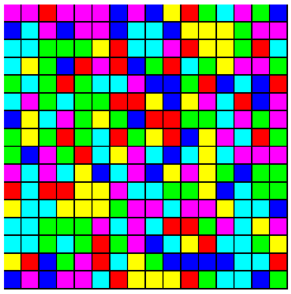
Hence, neighboring spins corresponding to similar pixels tend to align

Ferber and Worgotter. 1998

Using Potts segmentation to cluster images

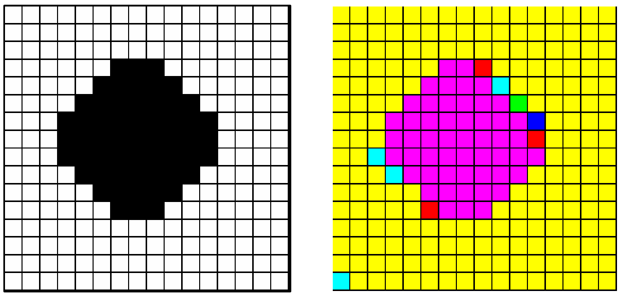
- 50 iterations. Spin States = 6. Start from random spin ("color") configuration.





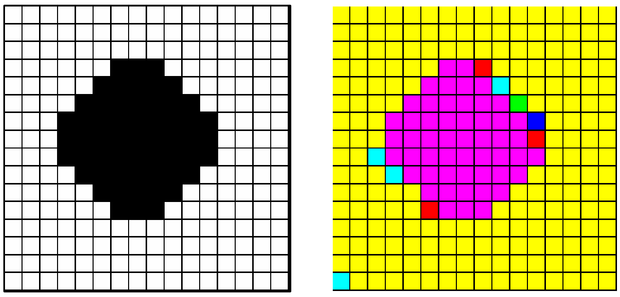
Using Potts segmentation to cluster images

- 50 iterations. Spin States = 6. Start from random spin ("color") configuration.

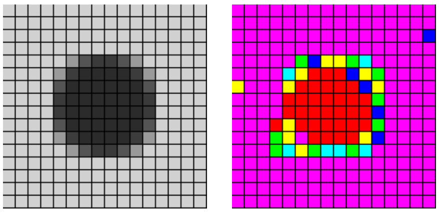


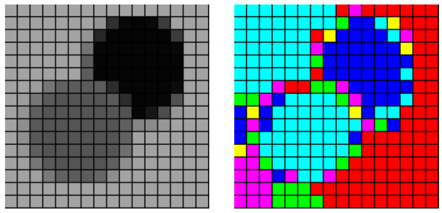
Using Potts segmentation to cluster images

- 50 iterations. Spin States = 6. Start from random spin (“color”) configuration.

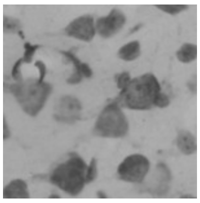


- other examples...



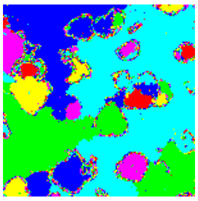


Neuron recognition using Potts clustering



150 x 150 = 2^{15} pixels

$\theta = 1$
$\beta = 1$



final state

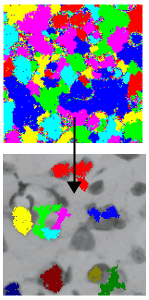
$$J_{ij} = 1 - \frac{|g_i - g_j|}{\theta \langle g_i - g_j \rangle}$$

$$P_{freeze} = (1 - e^{-\beta J_{ij}})\, \delta_{\sigma_i \sigma_j}$$

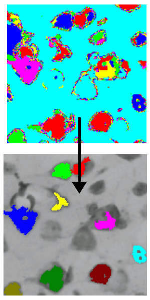
$$P_{flip} = e^{-\beta \Delta H}$$

Parallelization: create final states of different θ,β parameters (ex: 10 choices)

#1
$\theta = 3.8$
$\beta = 0.68$

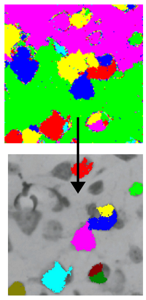


#2
$\theta = 1.8$
$\beta = 1.8$



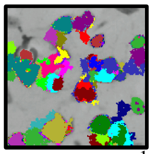
• • •

#10
$\theta = 4.6$
$\beta = 0.82$



• delete very large & small *clusters*:

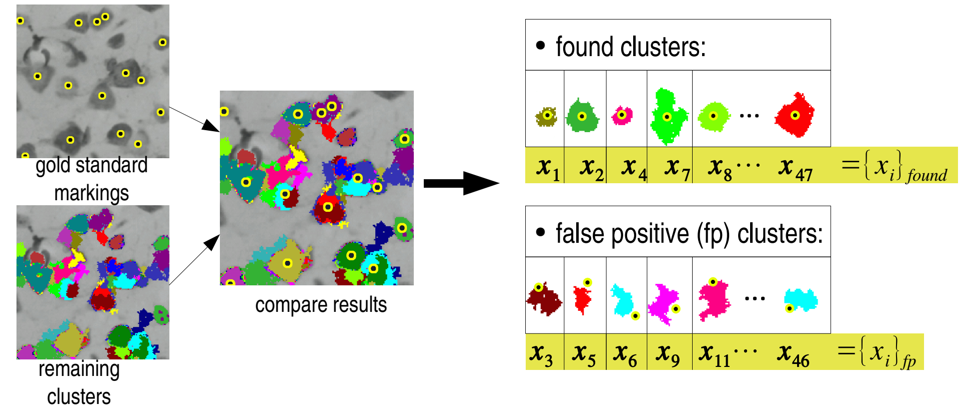
• overlay remaining clusters (ex: 47 total)



Peng et al. 2003

Cluster selection by computer training

- Make image a “gold standard” : a tissue sample marked by a neuroanatomist



example cluster

$$x_i = \begin{pmatrix} \text{area} \\ \text{avg. grayscale} \\ \text{grayscale variance} \\ \text{grayscale 'gyration'} \\ \text{gyration} \\ \text{edge length vs. area} \end{pmatrix}$$

- use $\{\boldsymbol{x}_i\}_{found}$ to find probability distribution $P_{found}(\boldsymbol{x})$
- use $\{\boldsymbol{x}_i\}_{fp}$ to find probability distribution $P_{fp}(\boldsymbol{x})$

Gaussian Mixture Models. (Dempster et al. 1977)

Finding neurons for any image (...that looks like the "gold standard")

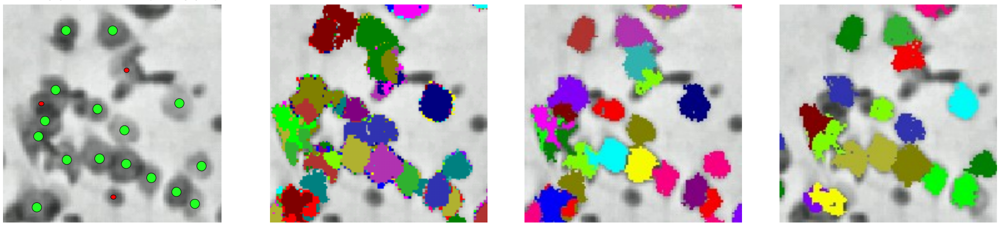
1. overlay final states from θ/β runs.
2. if cluster i is very small or very large, then DELETE.
3. if $P_{fp}(x_i) > P_{found}(x_i)$ then DELETE cluster i.
4. if x_i and x_j ($i \neq j$) overlap by > 0.8, and $P_{found}(x_j) > P_{found}(x_i)$, then DELETE cluster i.

<u>remaining clusters represent neurons.</u>

ex: two clusters compete for the same feature

an example...

• neuron • not neuron

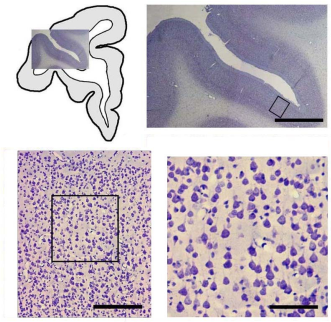


initial image | after step **2.** | after step **3.** | after step **4.** (**FIN)**

Method validation

Data Set

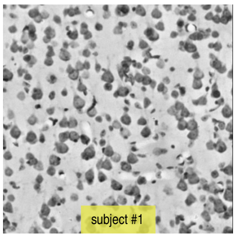
- prefontal cortex (area 46, layer 3) from Rhesus Monkey. 15 subjects of varying age, 8 pictures each = 120 total images.
- each image $2^9 \times 2^9$ (512x512) pixels with resolution 1 μm per pixel length, ~150 neurons per image.
- previously marked using semiautomated methods. Used for neuron-neuron correlation density map analysis (*Cruz. et al 2004.)*
- Randomly select 7 out of 15 subjects for analysis

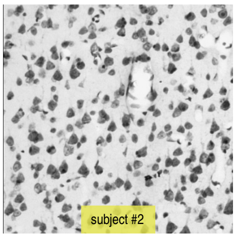


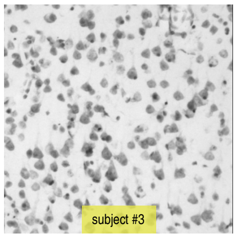
scale b=2.5mm, c=250 microns, d=100 microns

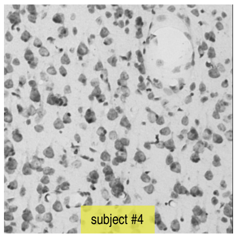
image courtesy of Daniel Roe

4 (out of 7) examples of variability of images type between subjects:









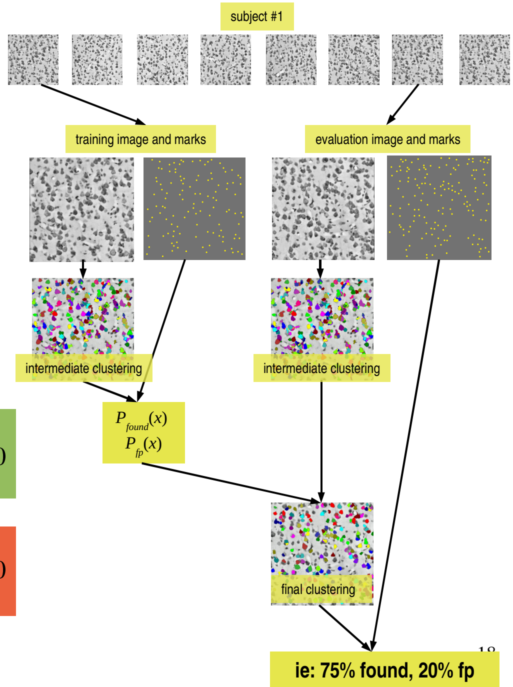
Method validation

For each subject:

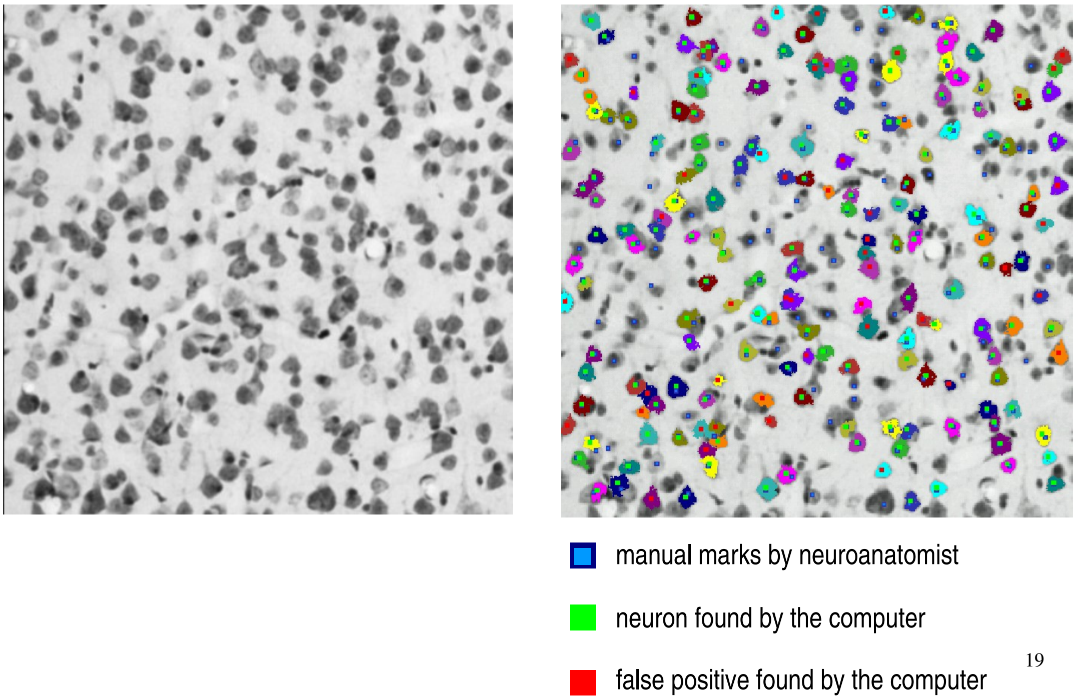
- randomly select *training* image and *evaluation* image from original 8 pictures.
- Neuroanatomist marks both images.
- use *training* image and marks to create $P_{found}(x_i)$ and $P_{fp}(x_i)$.
- analyze evaluation image.
- Test *evaluation* image cluster results compared to marks:

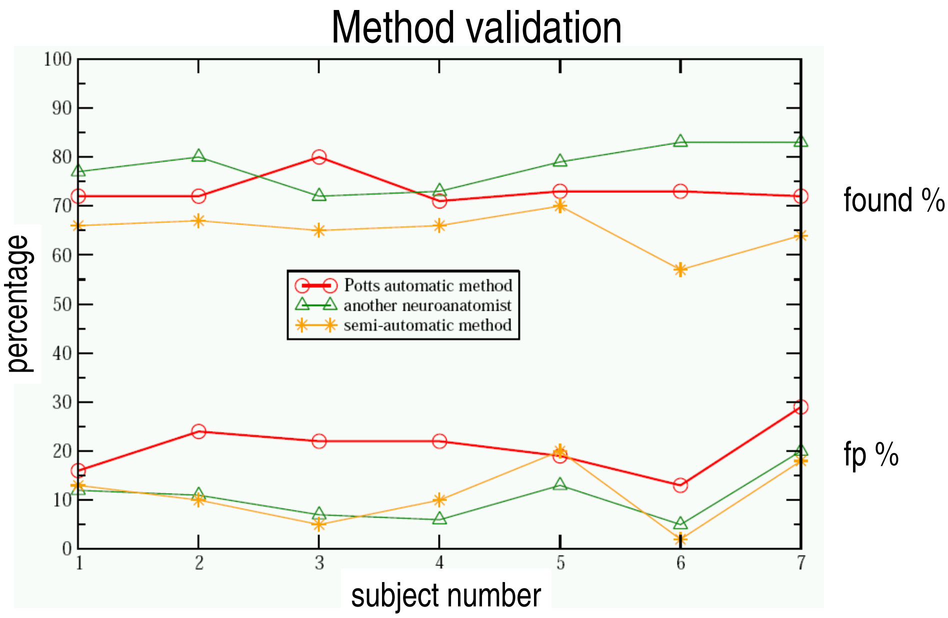
found %: (efficiency %) $\dfrac{no.\ of\ neurons\ found}{no.\ of\ gold\ standard\ neurons} \cdot 100$

fp %: (100- purity %) $\dfrac{no.\ of\ false\ positives}{no.\ of\ gold\ standard\ neurons} \cdot 100$



Method validation





Potts:	found : 73±3 %	fp: 21±5%
another neuroanatomist:	found : 78±4 %	fp: 11±6%
semi-automatic method:	found : 65±4%	fp: 11±6%

Conclusions so far

A combination of using:

- Potts segmentation
- parallelization, and
- a trained network

allows for determination of individual neuron location in diverse tissue samples with the desired accuracy.

Next Steps

- Replace Potts clustering with quicker more efficient clustering method.
- Apply the method to correlative measurements.
- Apply the method to larger samples not possible for manual marking.
- Extend the method to measure neuron size, shape, and location within a 3-dimensional framework .
- Explore other trained network models in order to reduce recognition error.