Machine Learning in Image Processing

Quantitative Big Imaging Course 2015

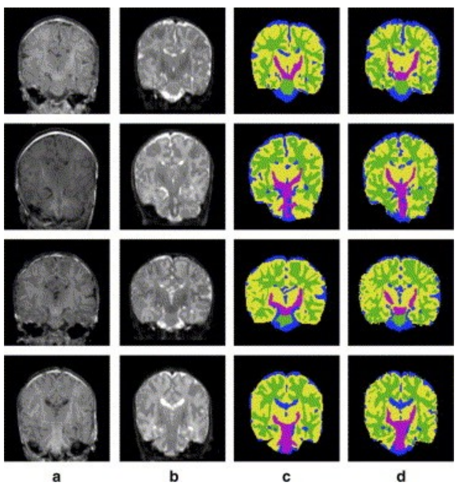
Thursday March 19th, 2015

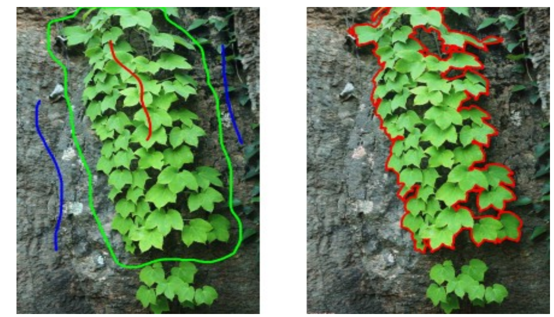
Lecturer: Aurelien Lucchi
Post-doc at the Institute of Machine Learning

data analytics lab

Image Segmentation

- Goal: partition an image into meaningful **regions** with respect to a particular application.





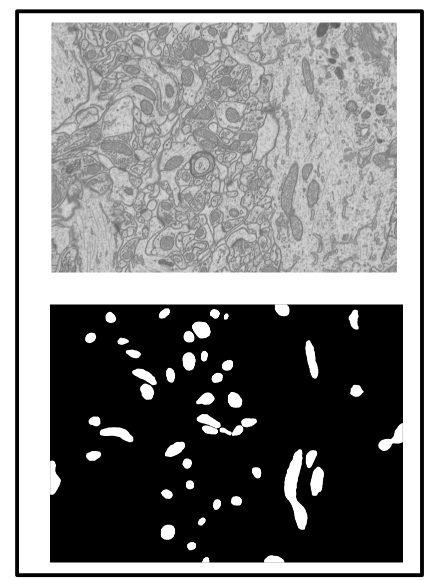
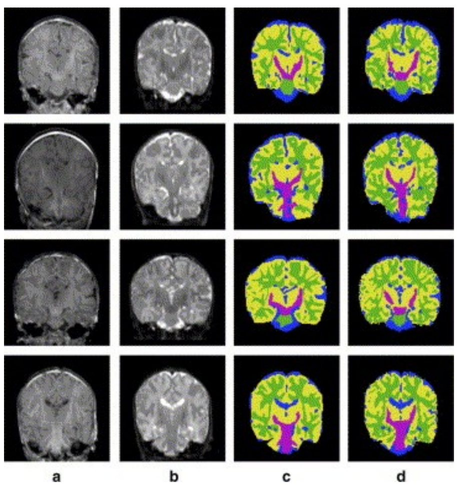
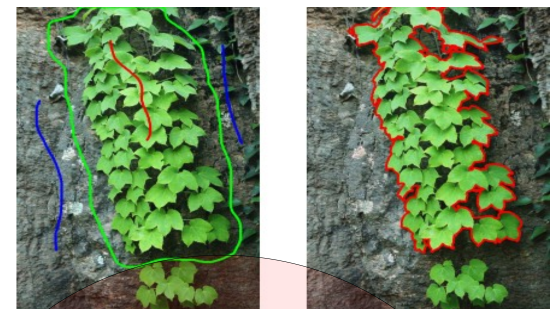
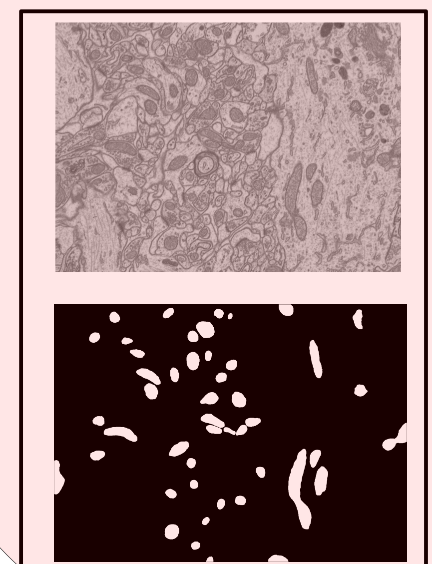


Image Segmentation

- Goal: partition an image into meaningful **regions** with respect to a particular application.







ETH zürich

Image Segmentation

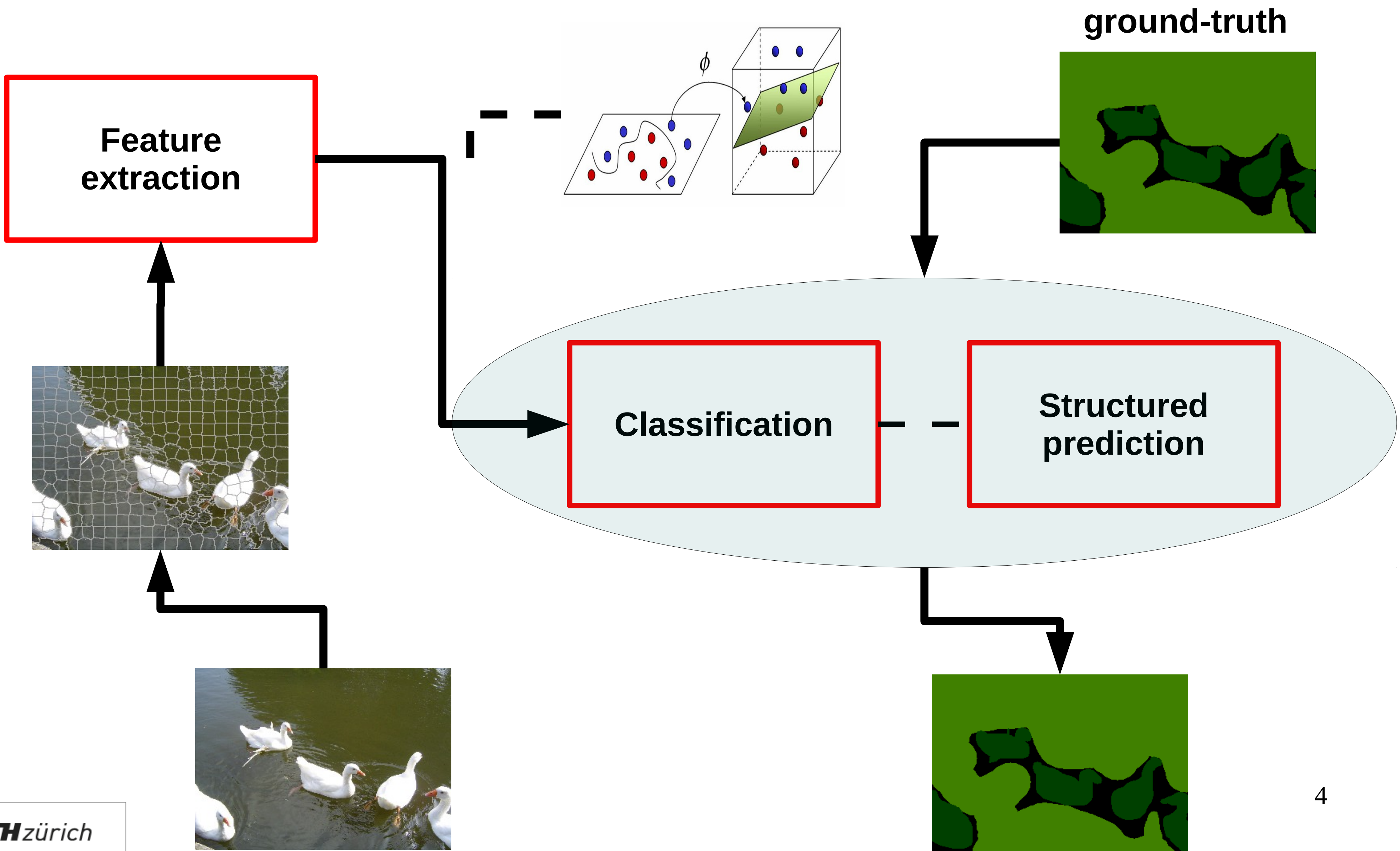
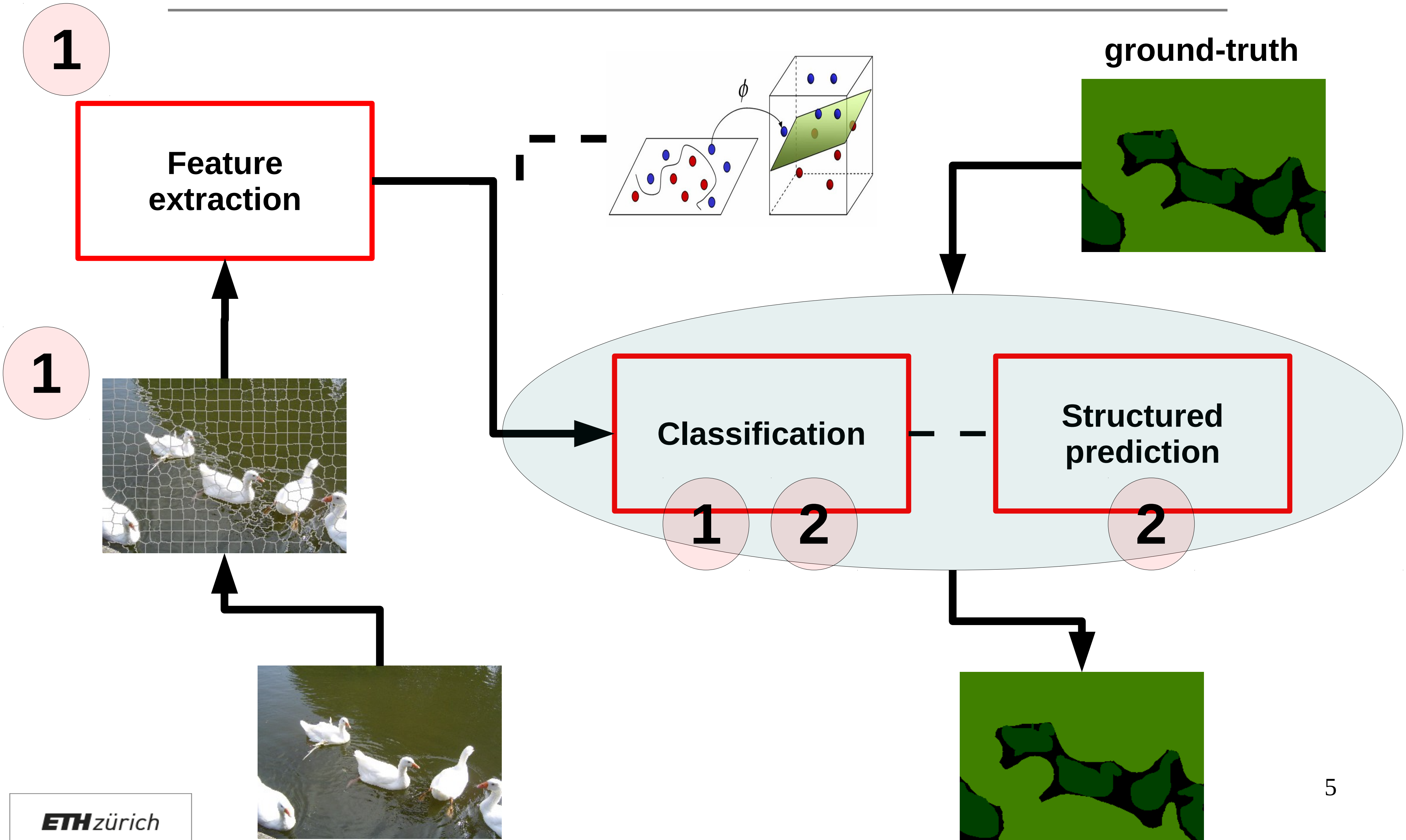


Image Segmentation



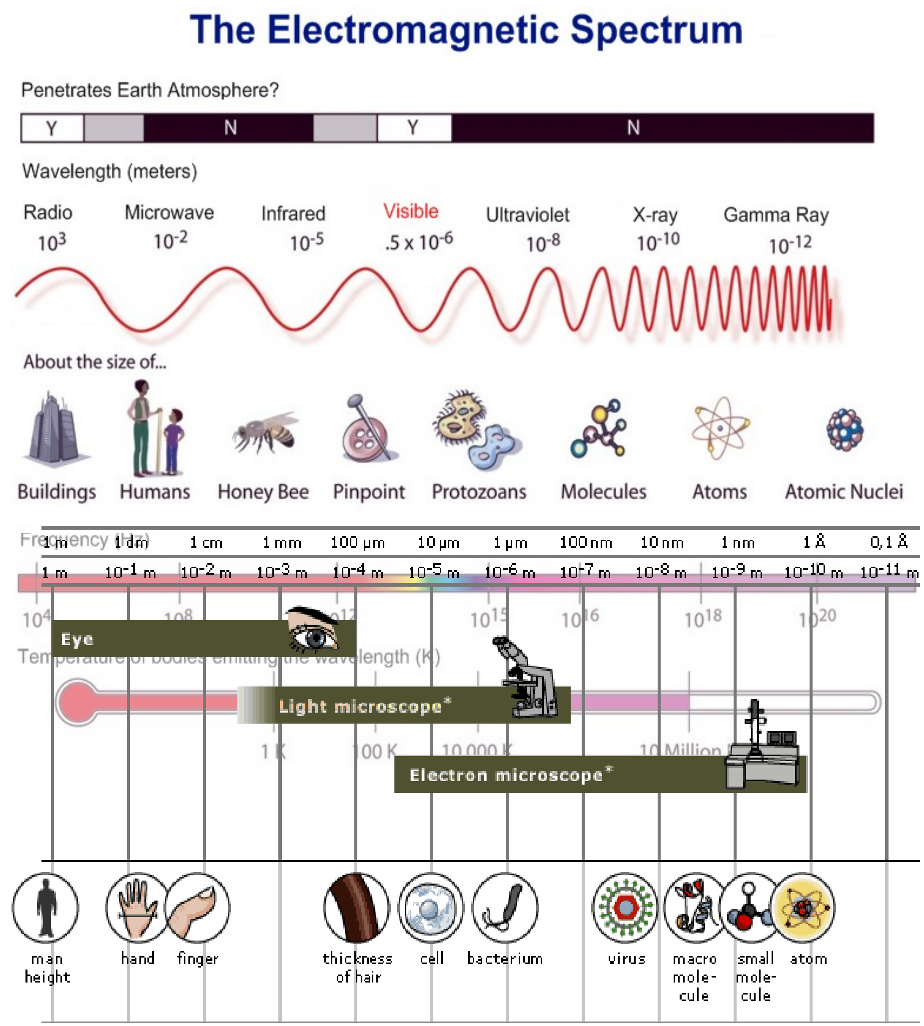
ETH zürich

Outline

1. Segmentation of Mitochondria in EM Images using a Support Vector Machine
2. Structured Prediction for Image Segmentation
    - A.k.a Conditional Random Fields

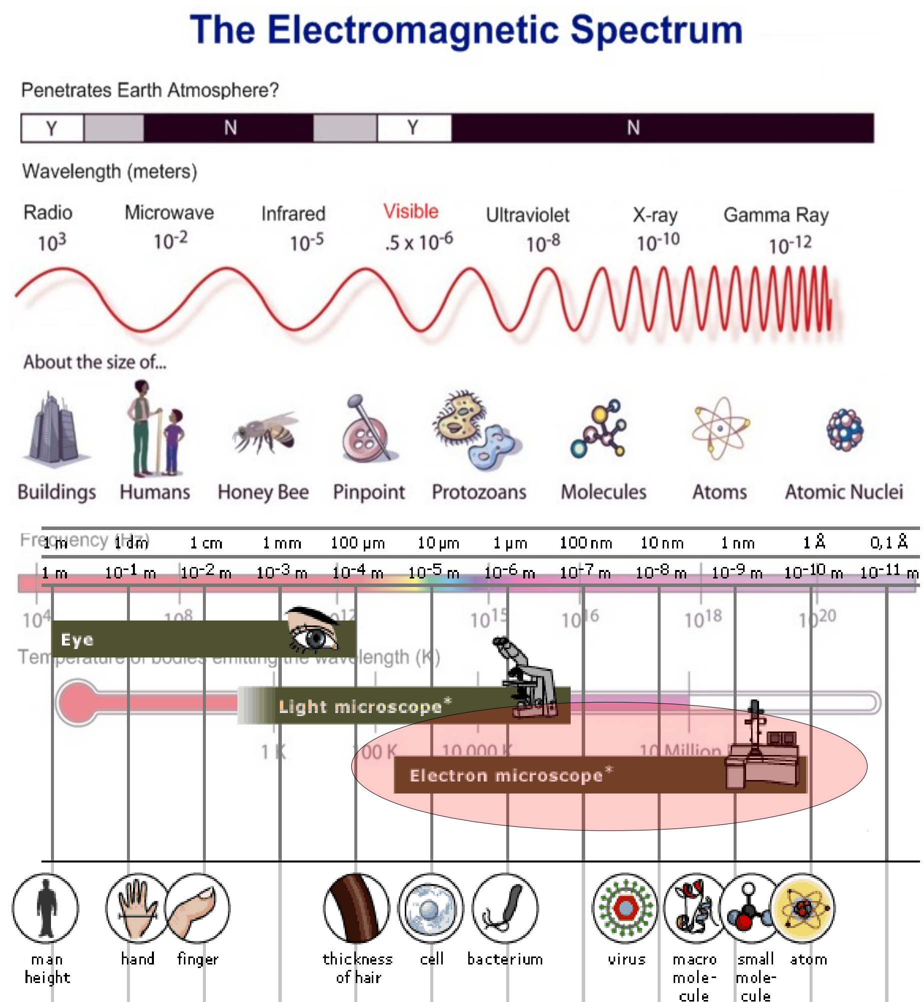
1. Segmentation of Mitochondria in EM Images using a Support Vector Machine

Understanding the Brain



ETH zürich

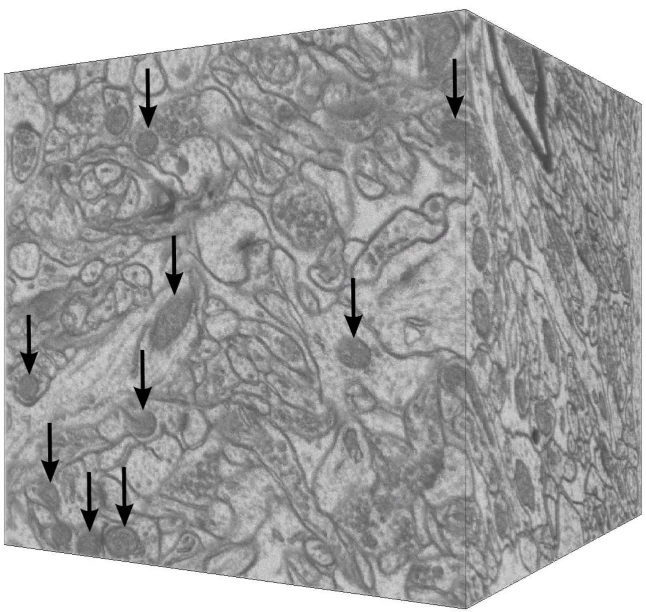
Understanding the Brain

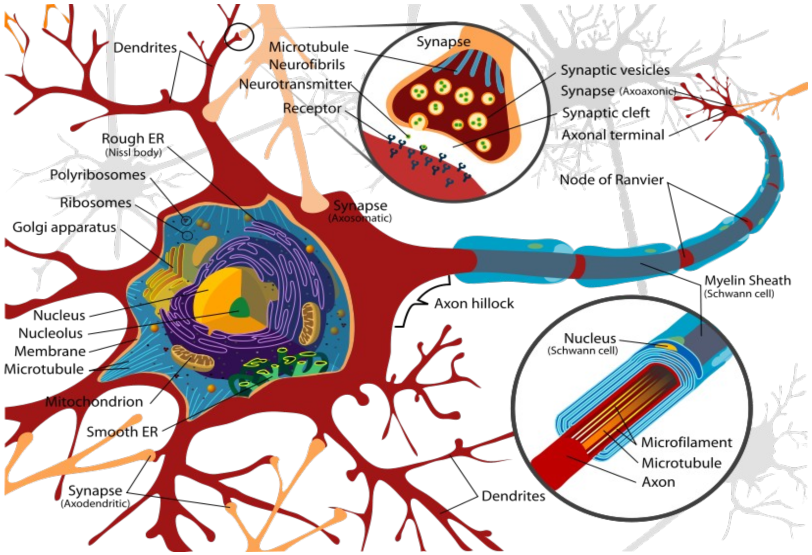


ETH zürich

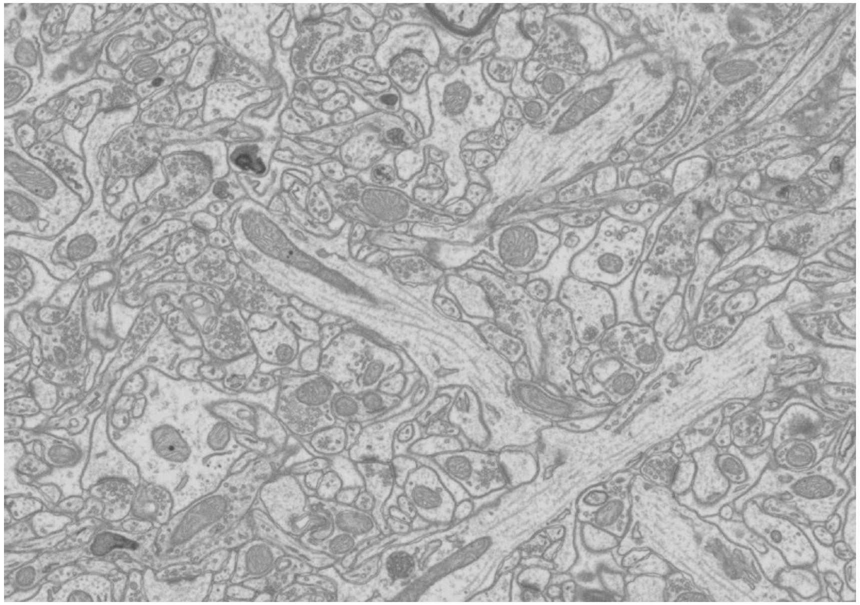
Electron Microscopy Data

- Human brain contains ~100 billion (10^{11}) neurons and 100 trillion (10^{14}) synapses.



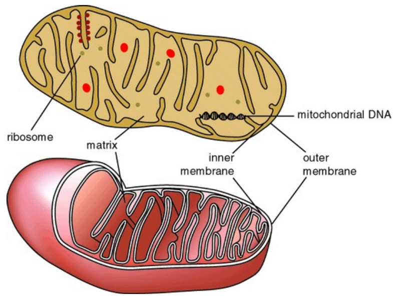


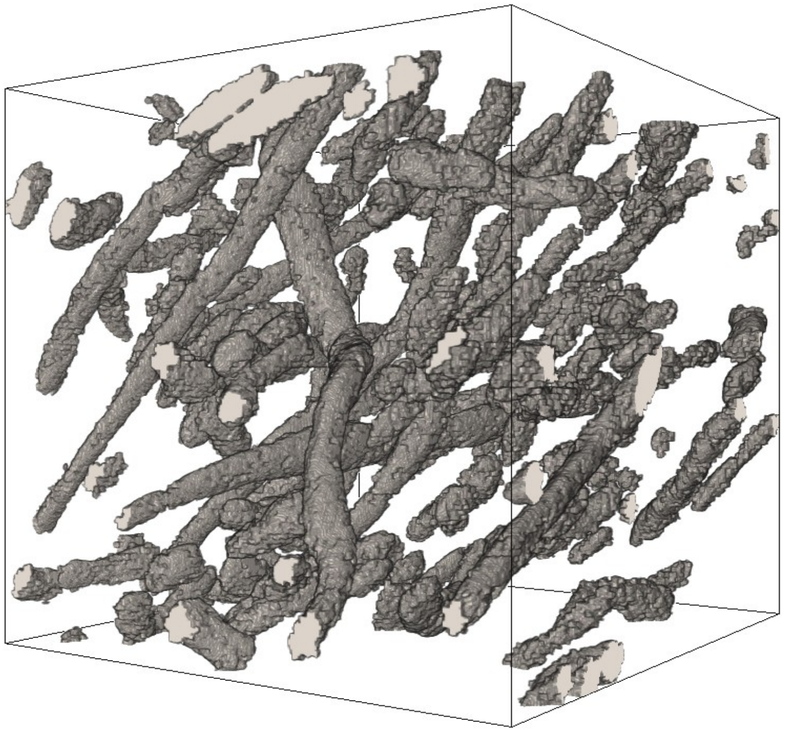
5 × 5 × 5 μm section taken from the CA1 hippocampus, corresponding to a 1024 × 1024 × 1000 volume (N ≈ 10^9 total voxels)

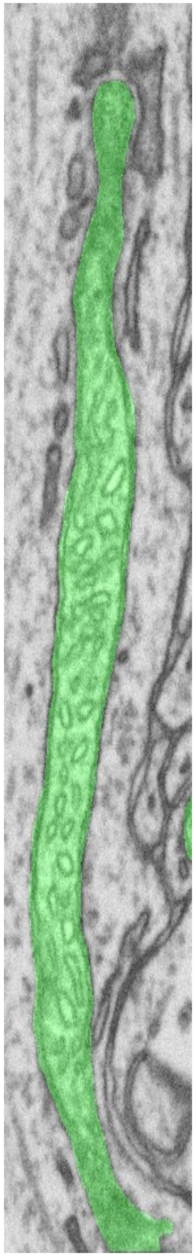


Mitochondria Segmentation

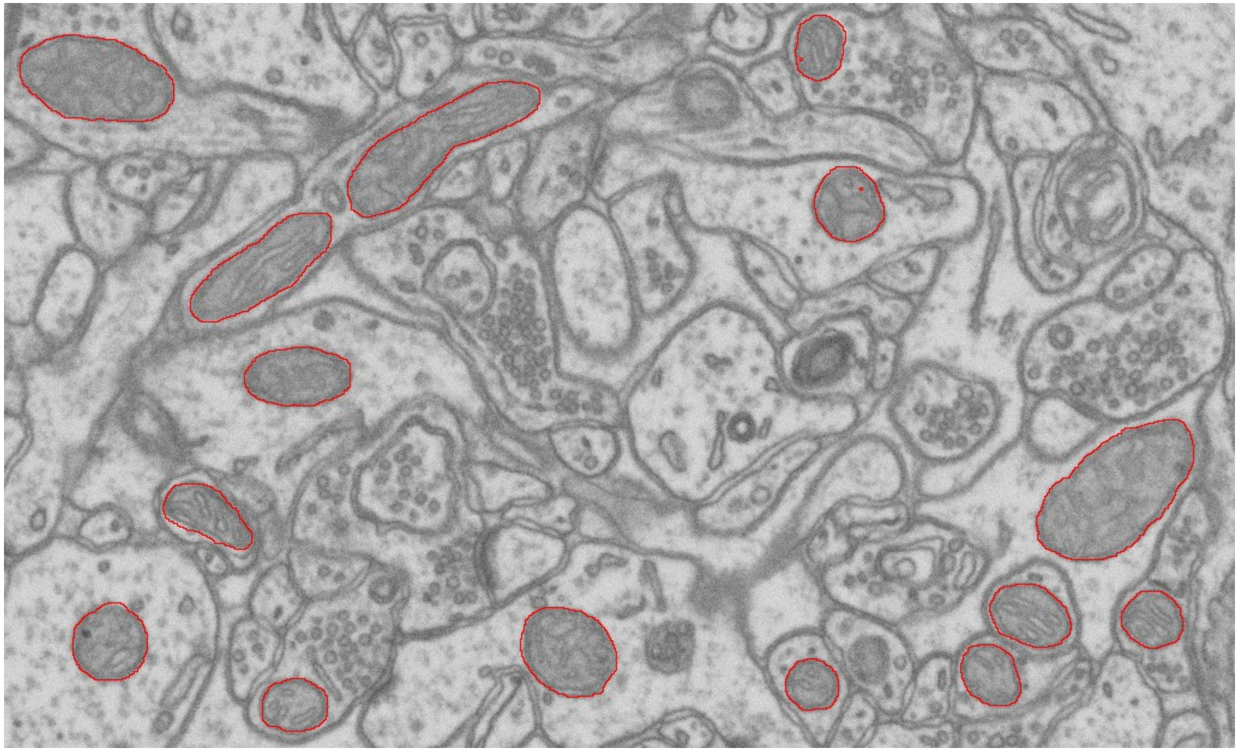
- **Difficulties** :
  - Vesicles and cell boundaries appear similar to mitochondria.
  - Assumptions about the **shape** are difficult.

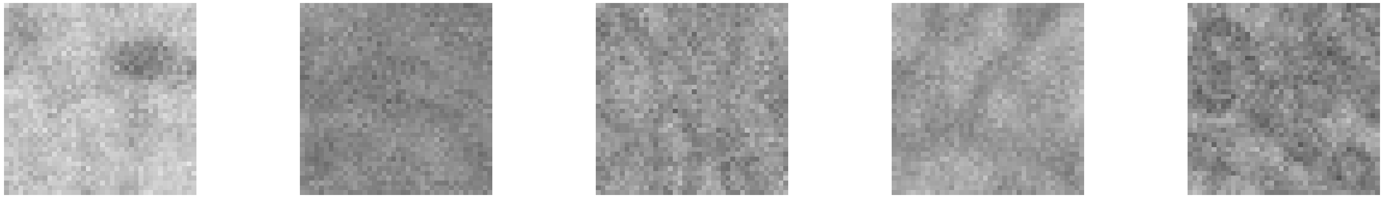




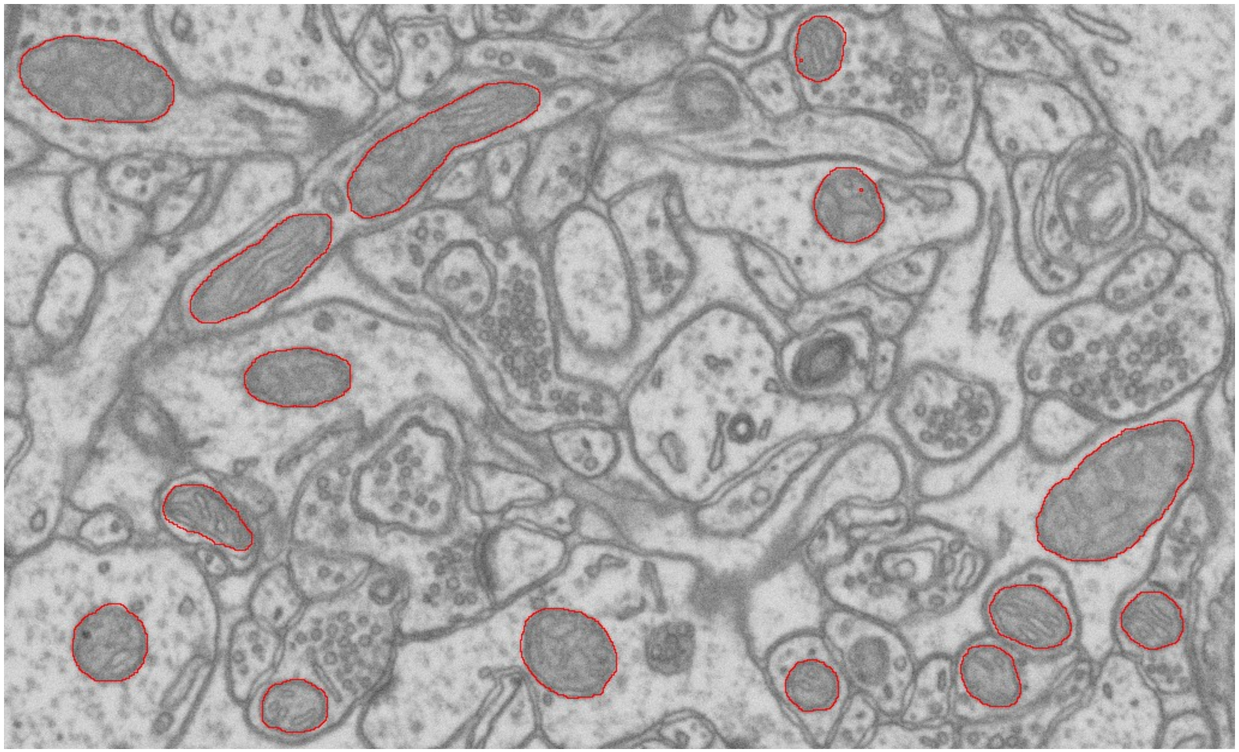


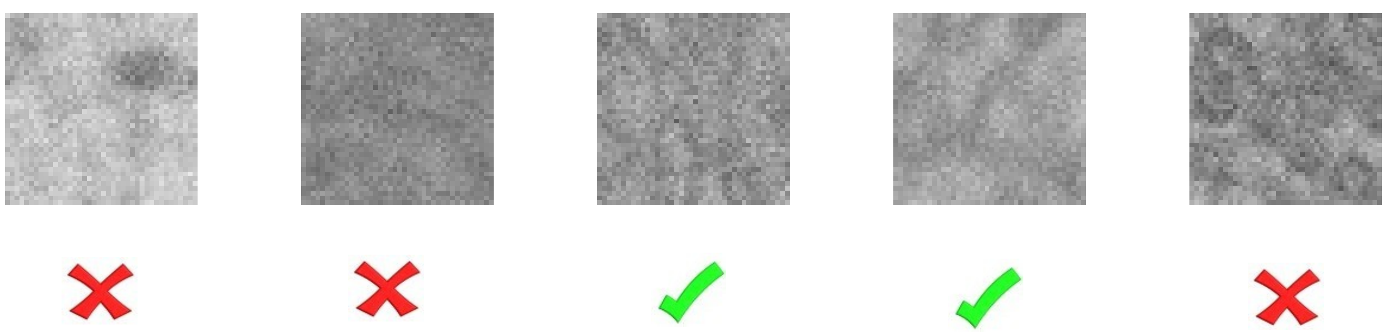
Mitochondria Segmentation



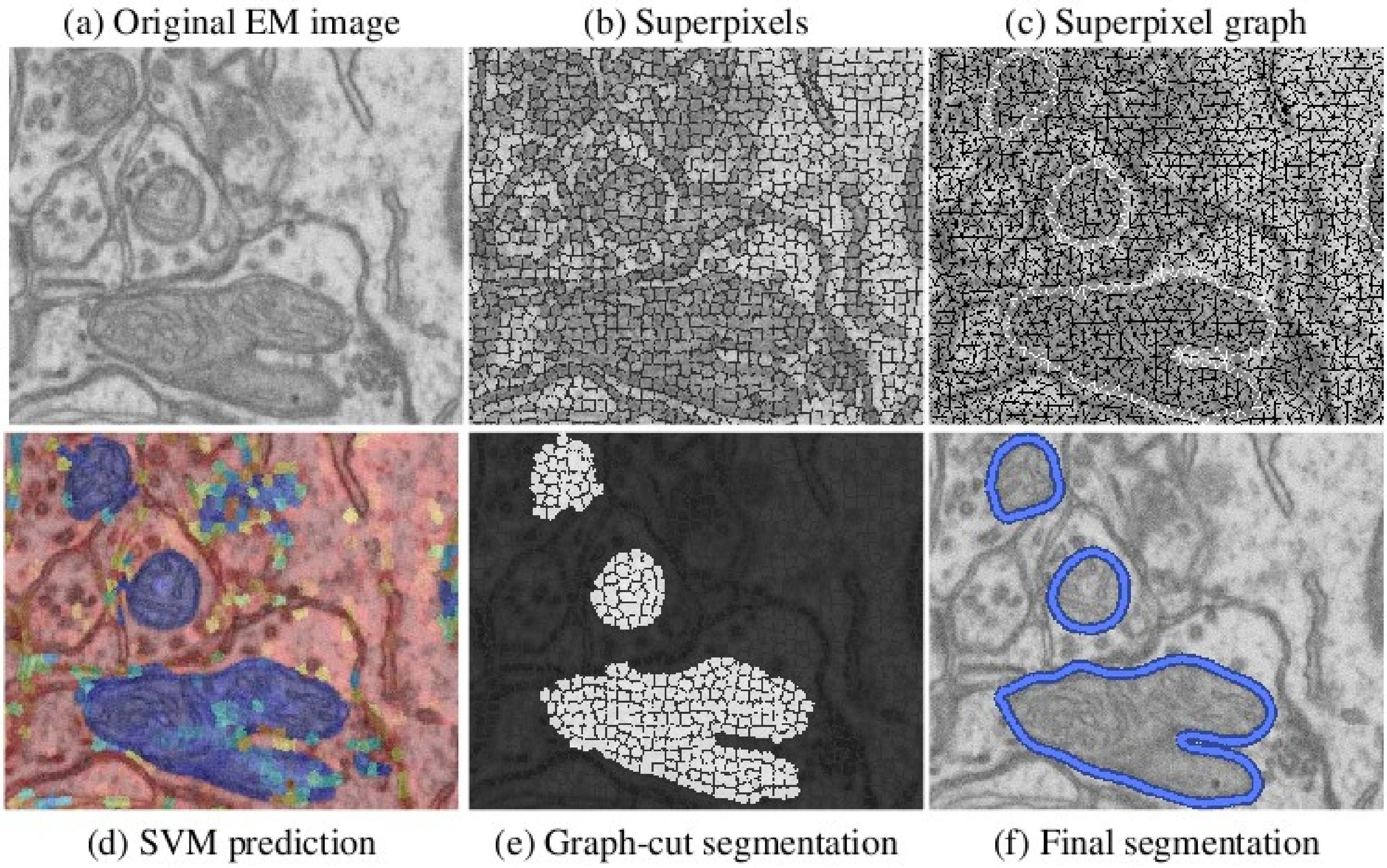


Mitochondria Segmentation



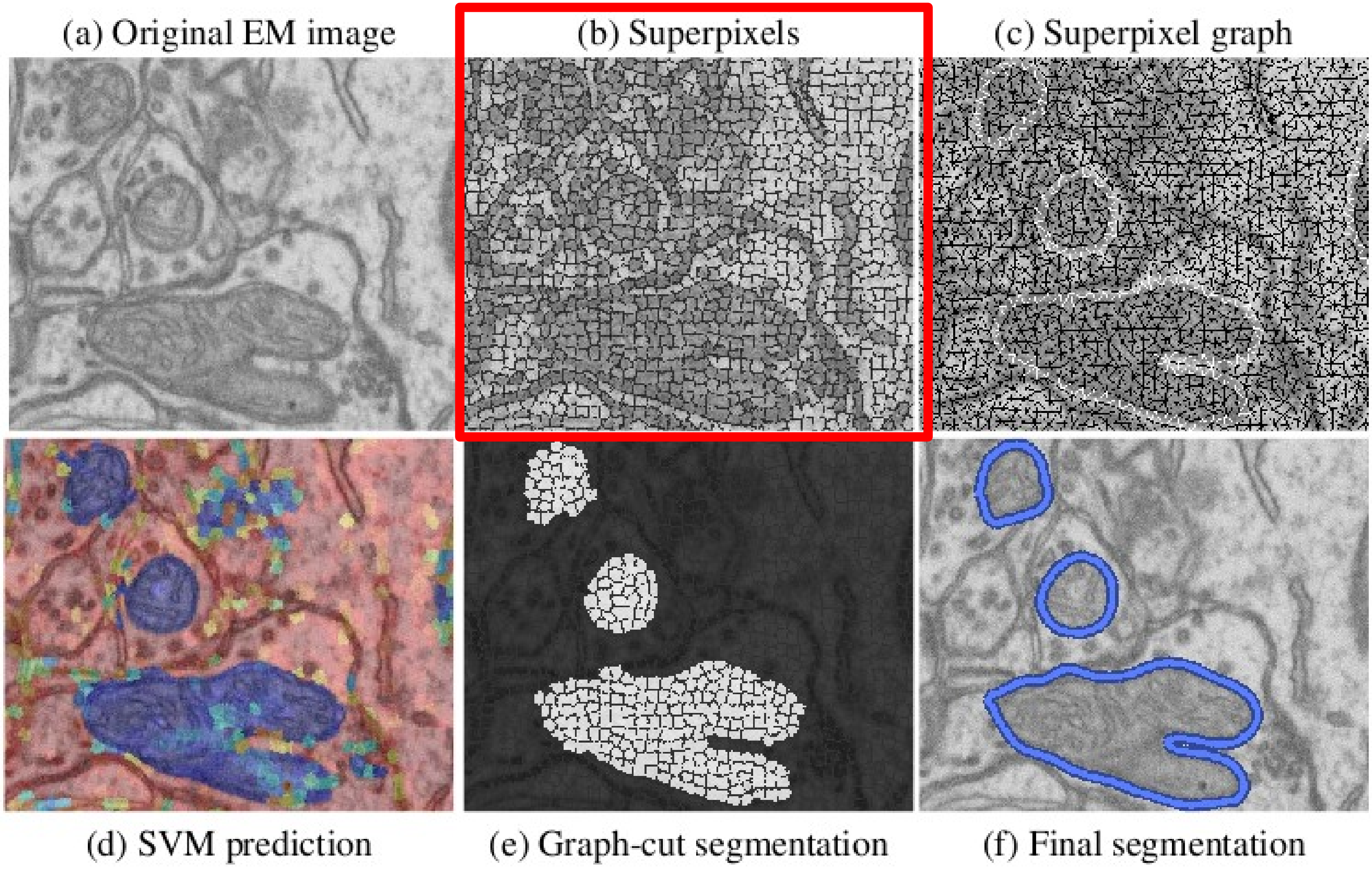


Approach



A Fully Automated Approach to Segmentation of Irregularly Shaped Cellular Structures in EM Images, MICCAI, Beijing, China, 2010, Lucchi et al.

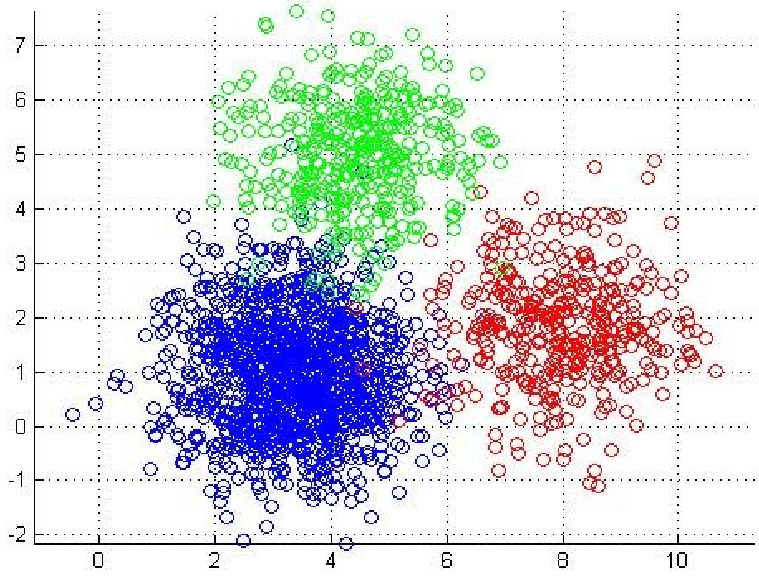
Approach



ETH zürich

SLIC Superpixels

- Clusters pixels in the combined five-dimensional color and image plane space.

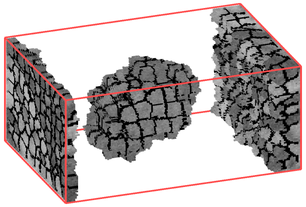


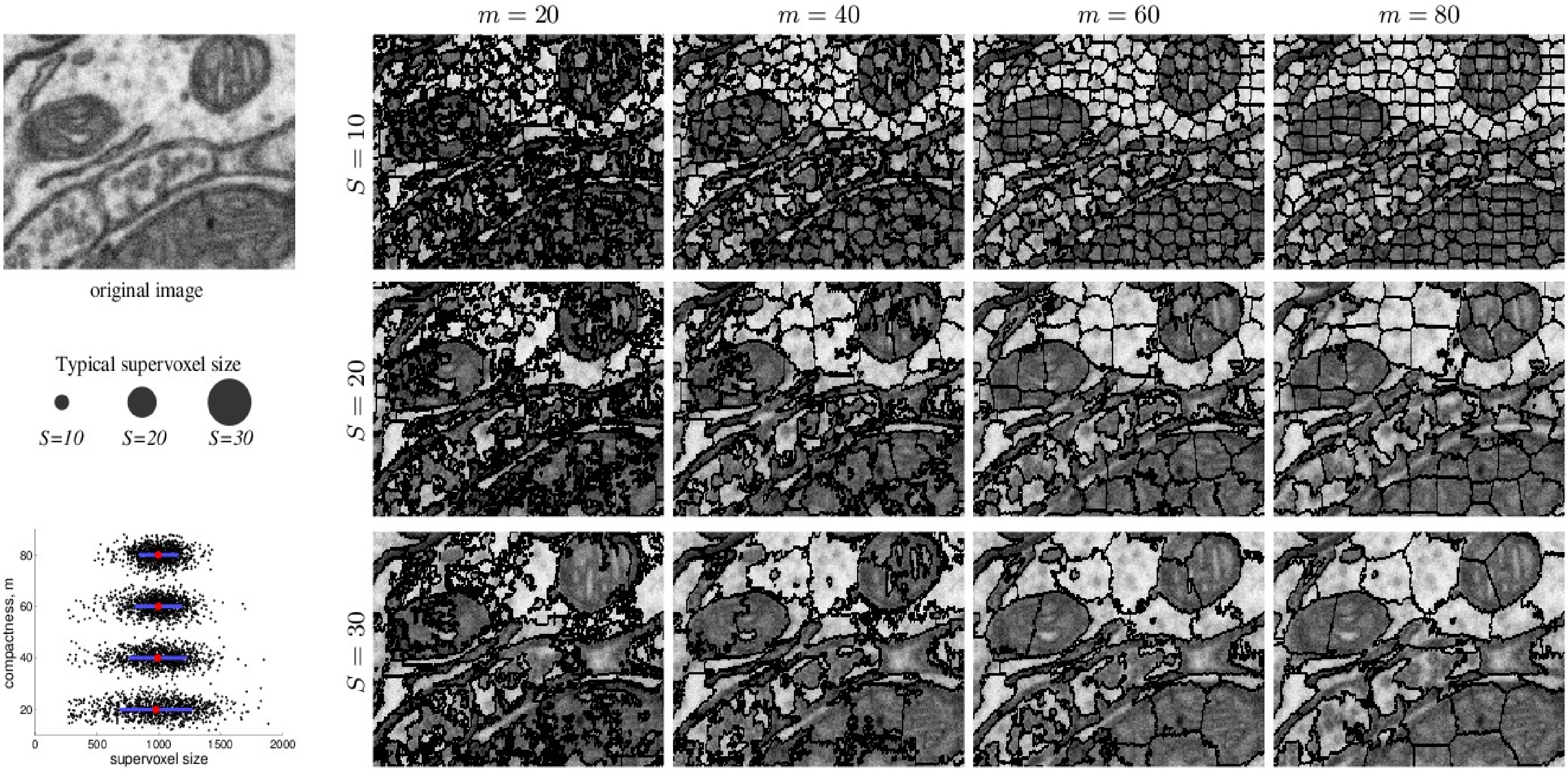
SLIC Superpixels

- Clusters pixels in the combined five-dimensional color and image plane space.
- Efficiently generate **compact**, nearly **uniform** superpixels.

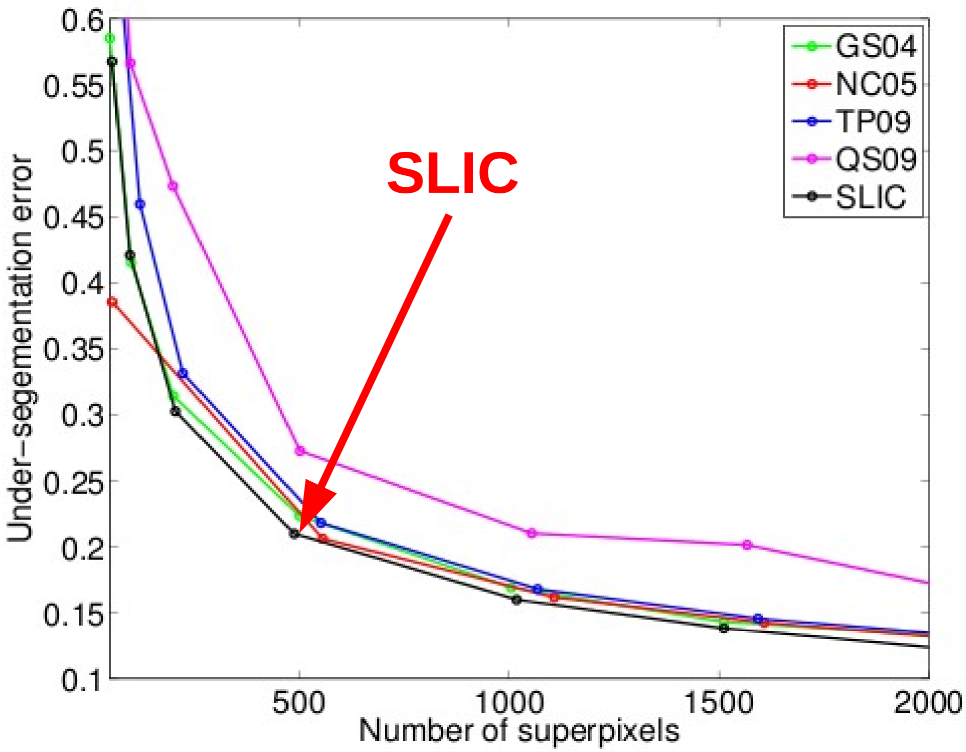
SLIC Superpixels Compared to State-of-the-art Superpixel Methods, IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI), 2012.
Source code available online.

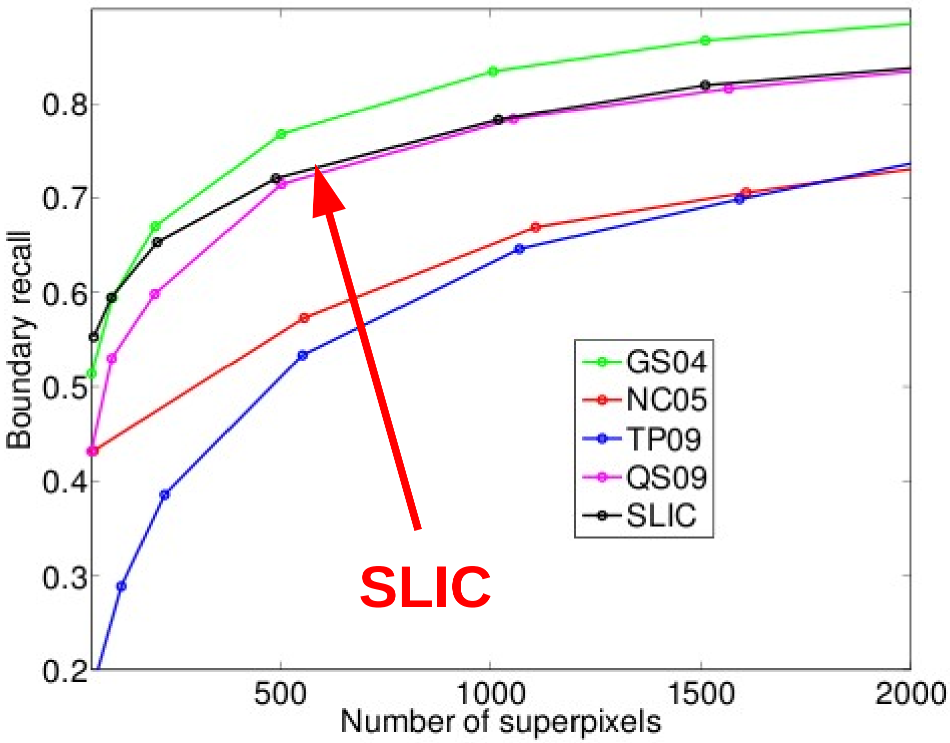
SLIC Supervoxels





SLIC Superpixels



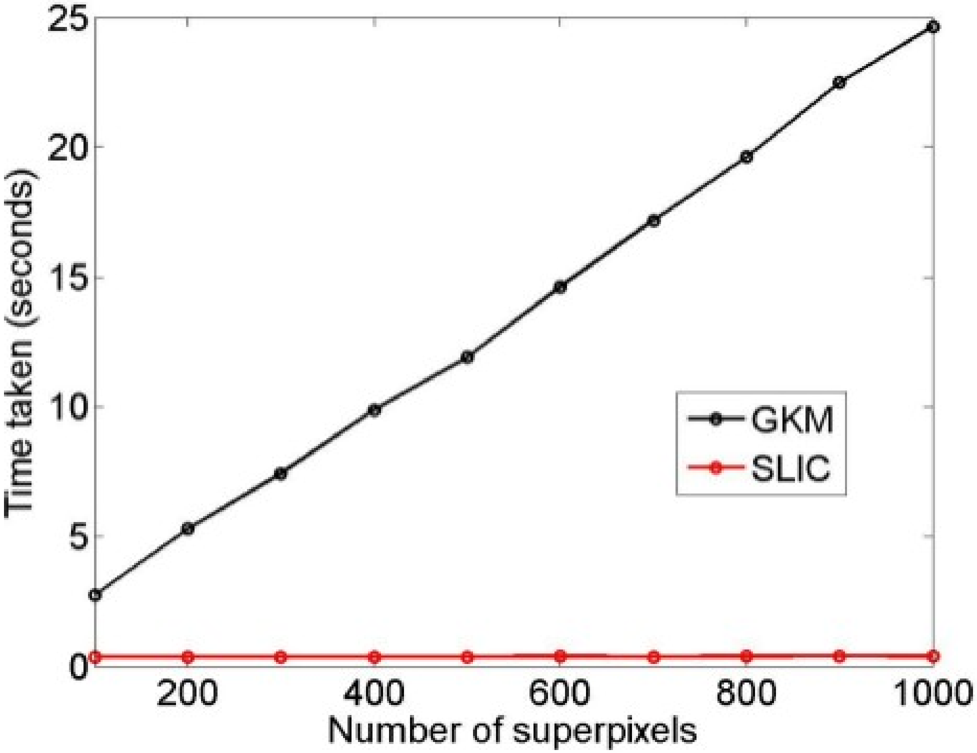


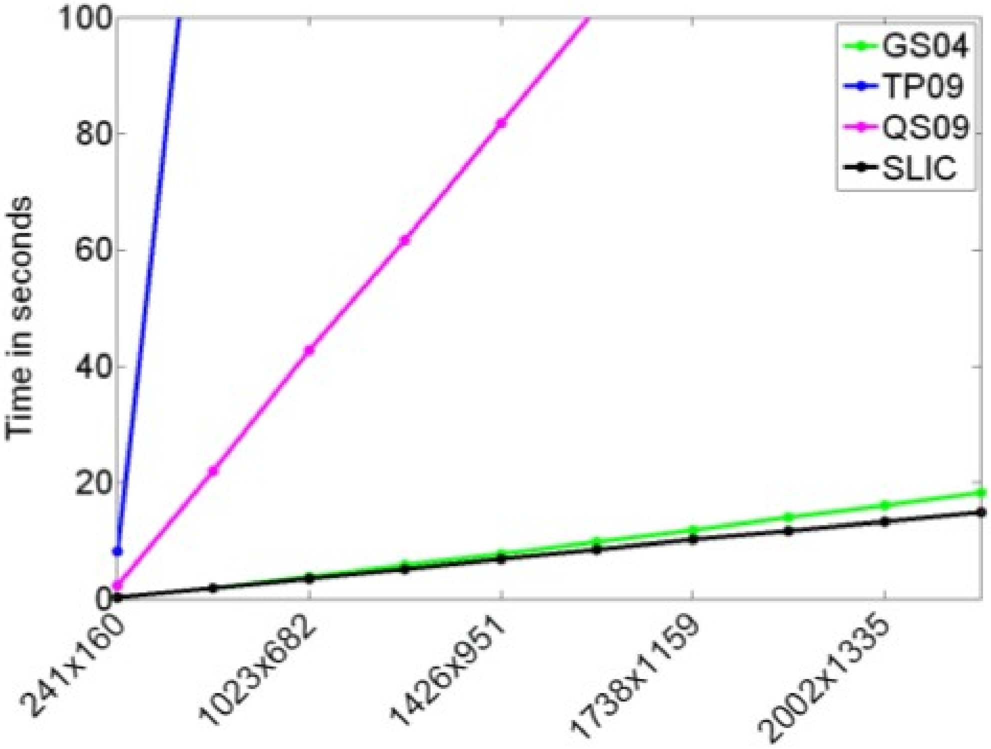
Under-segmentation error = error with respect to a known ground truth.

Boundary recall = fraction of ground truth edges fall within one pixel of a least one superpixel boundary.

- GS04: Efficient Graph-Based Image Segmentation, P. Felzenszwalb and D. Huttenlocher.
- NC05: Normalized Cuts and Image Segmentation, J. Shi and J. Malik.
- QS09: Quick Shift and Kernel Methods for Mode Seeking, A. Vedaldi et al.
- TP09: Turbopixels: Fast Superpixels Using Geometric Flows, A. Levinshtein et al.

SLIC Superpixels



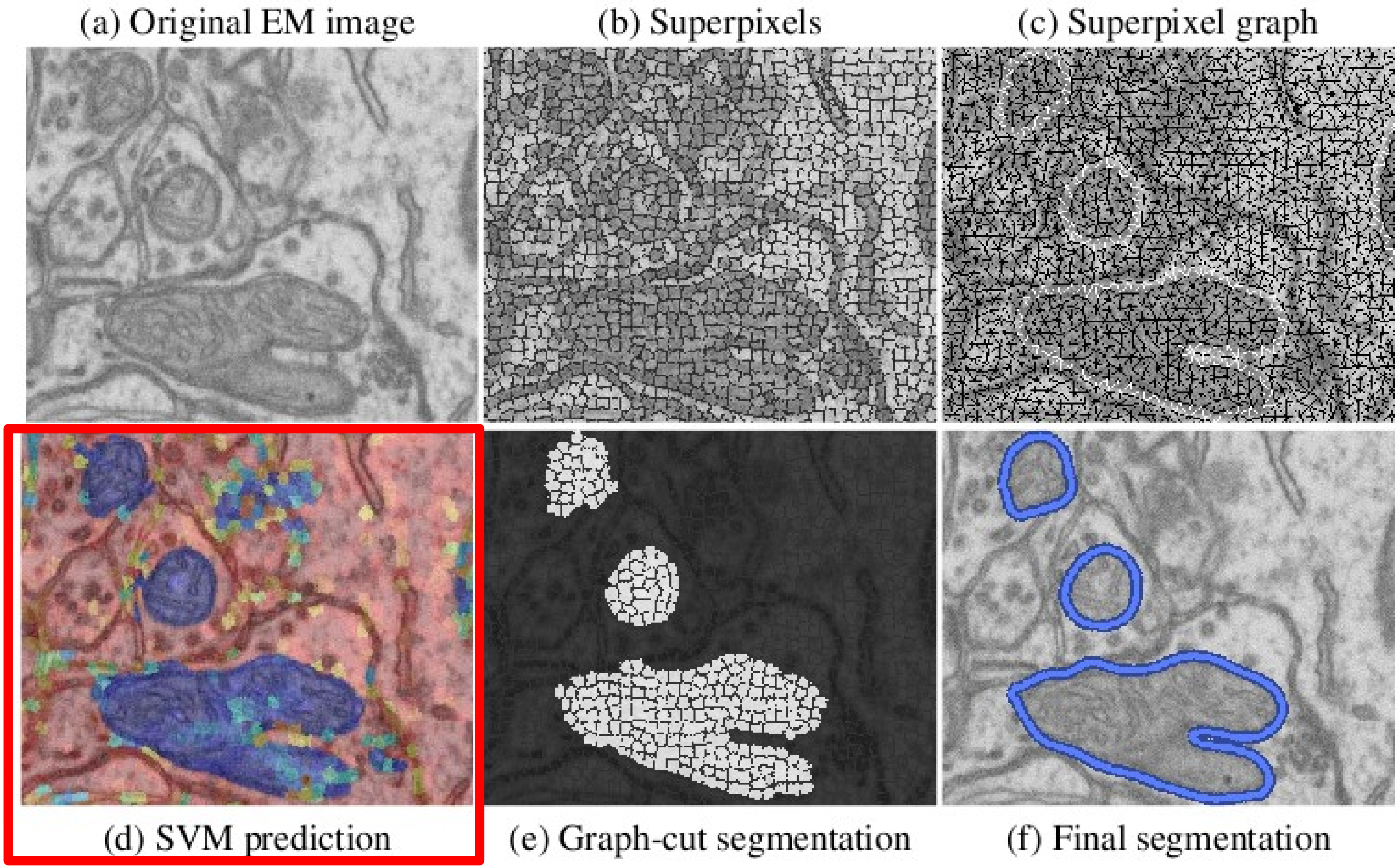


- GKM: 10 iterations of k-means.
- GS04: Efficient Graph-Based Image Segmentation, P. Felzenszwalb and D. Huttenlocher.
- NC05: Normalized Cuts and Image Segmentation, J. Shi and J. Malik.
- QS09: Quick Shift and Kernel Methods for Mode Seeking, A. Vedaldi et al.
- TP09: Turbopixels: Fast Superpixels Using Geometric Flows, A. Levinshtein et al.

SLIC Superpixels

- **Simple** approach, re-implemented in Fiji, vlfeat, scikit + GPU implementation.
- More info: http://ivrg.epfl.ch/supplementary_material/RK_SLICSuperpixels/index.html

Approach

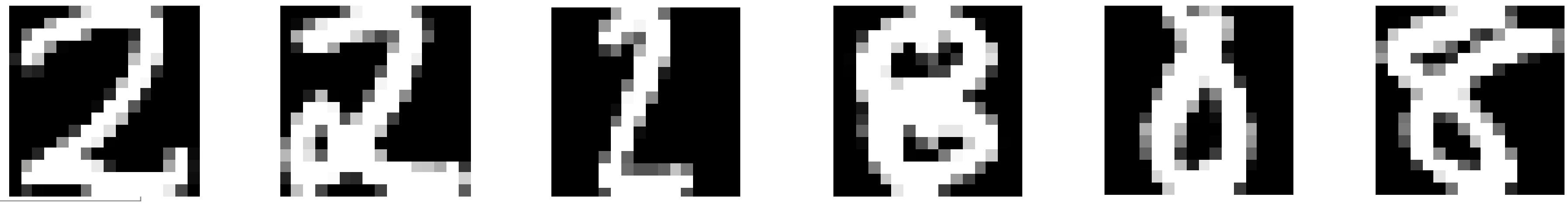

(a) Original EM image
(b) Superpixels
(c) Superpixel graph
(d) SVM prediction
(e) Graph-cut segmentation
(f) Final segmentation

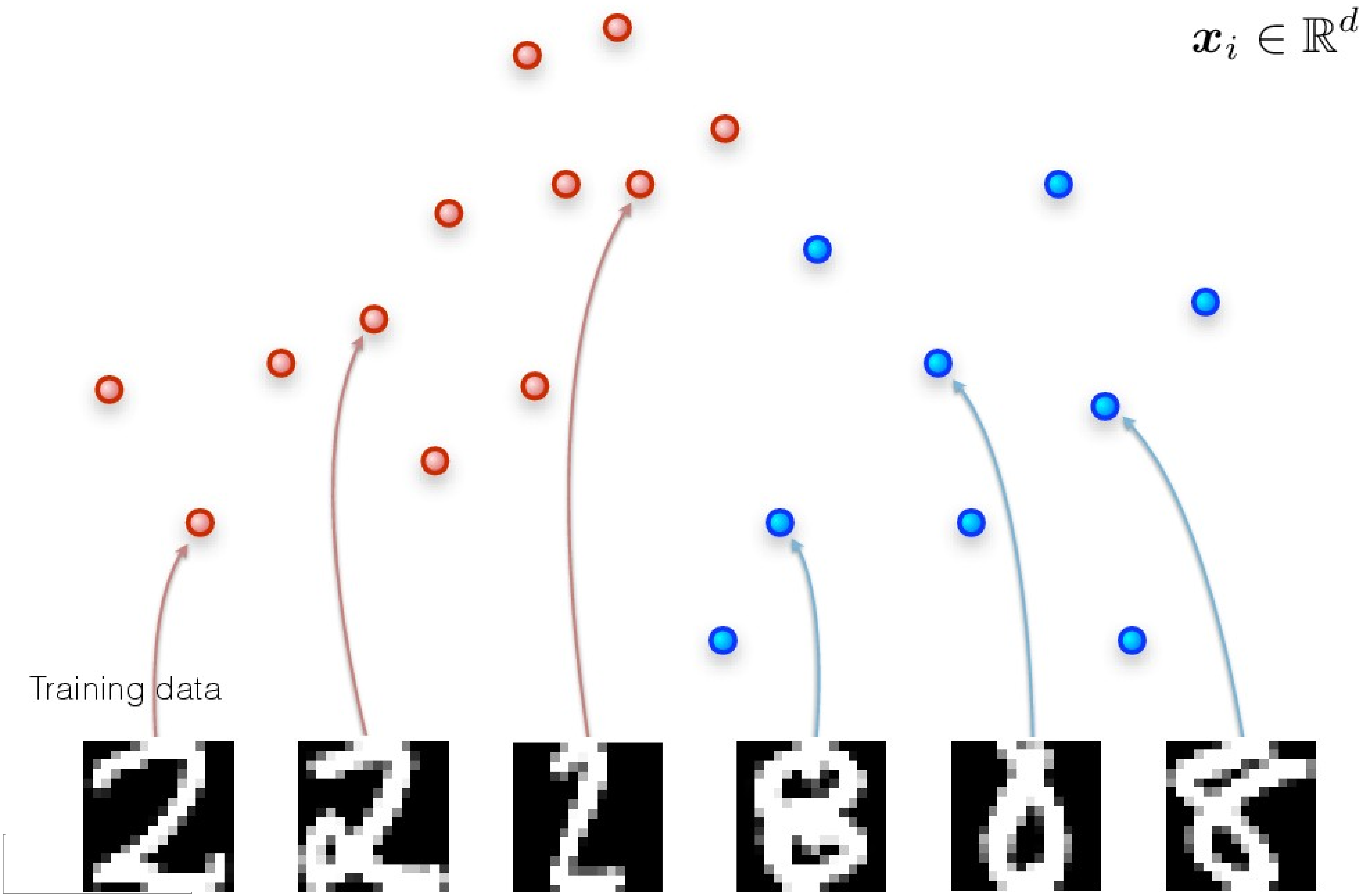
SVM

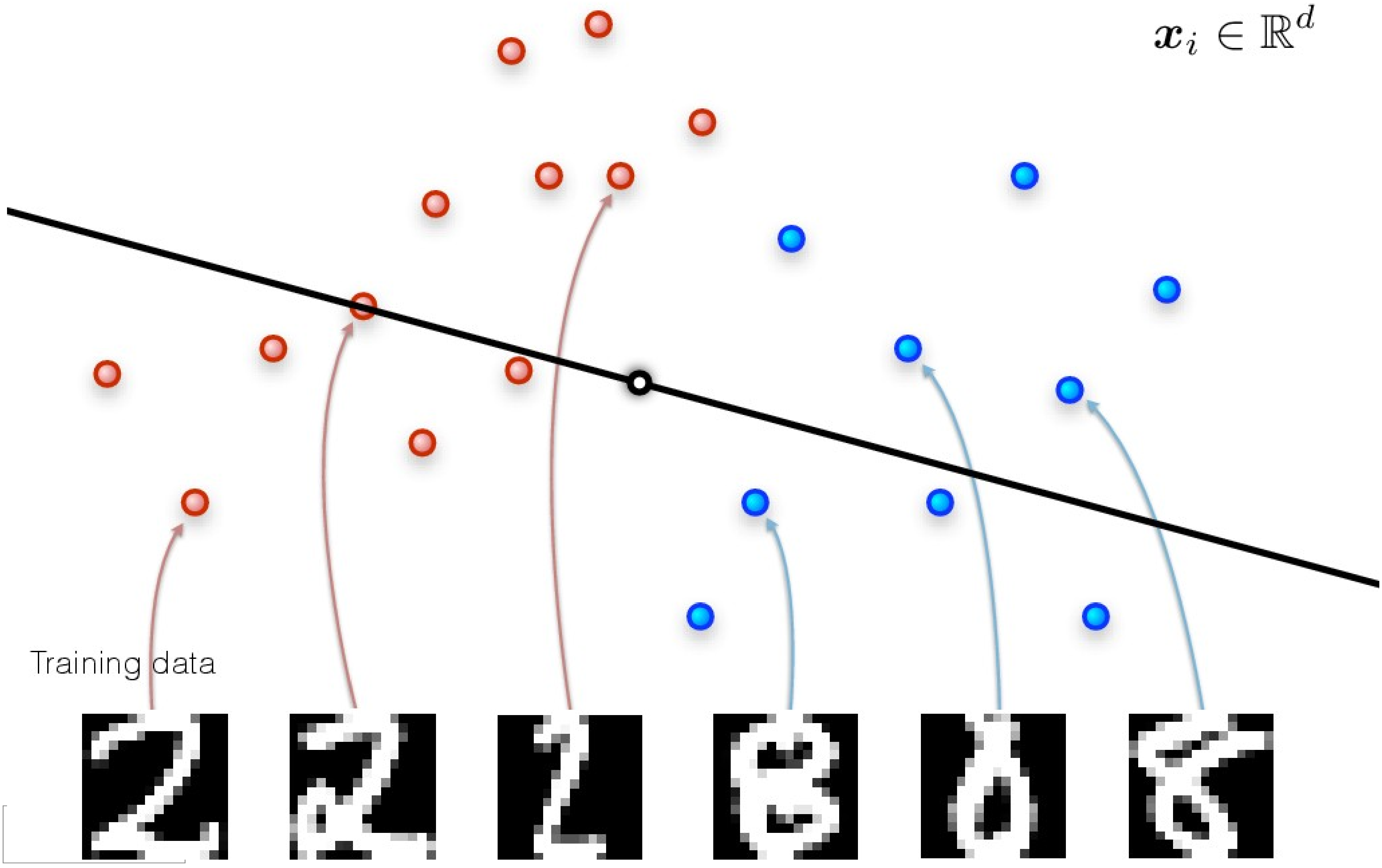
- “In machine learning, support vector machines (SVMs) are supervised learning models with associated learning algorithms that analyze data and recognize patterns, used for **classification** and regression analysis.” (wikipedia)
  - Supervised => need groundtruth
  - Recognize patterns => assume we can discriminate between data points that belong to different classes (e.g. cell vs background).

$$\boldsymbol{x}_i \in \mathbb{R}^d$$

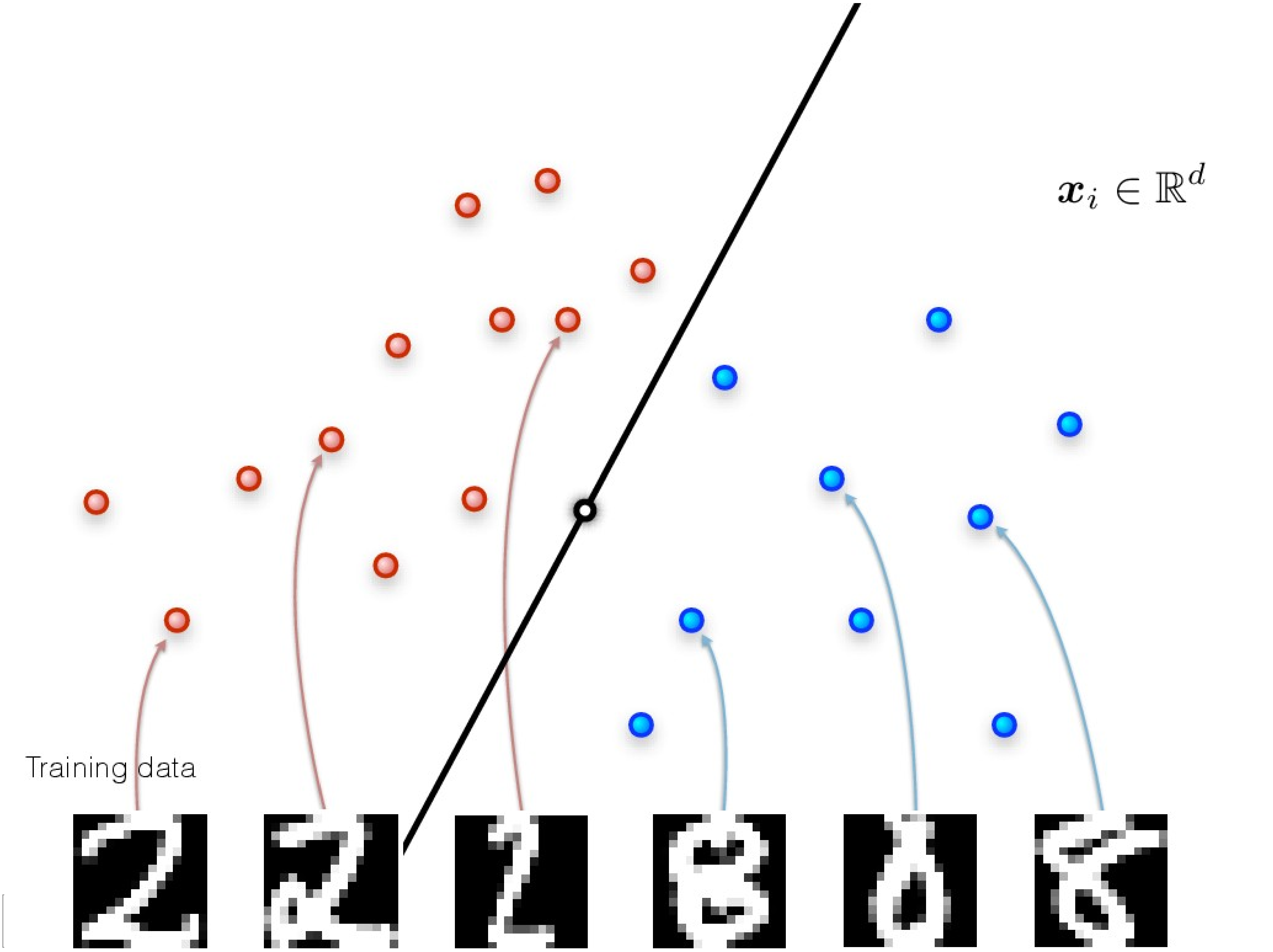
Training data



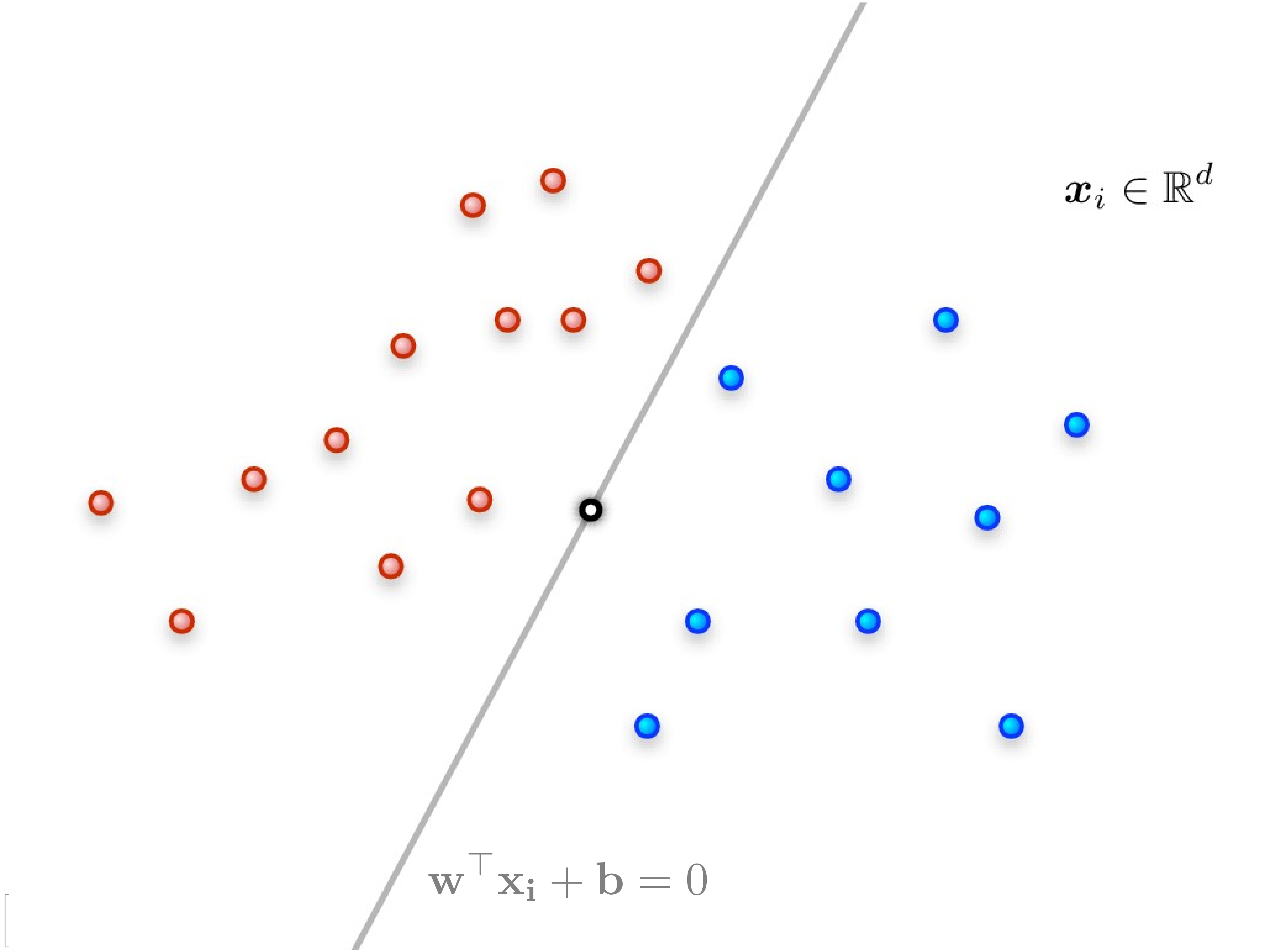

$x_i \in \mathbb{R}^d$
Training data



$\boldsymbol{x}_i \in \mathbb{R}^d$
Training data



$\boldsymbol{x}_i \in \mathbb{R}^d$
Training data


$\boldsymbol{x}_i \in \mathbb{R}^d$
$\mathbf{w}^\top \mathbf{x_i} + \mathbf{b} = 0$

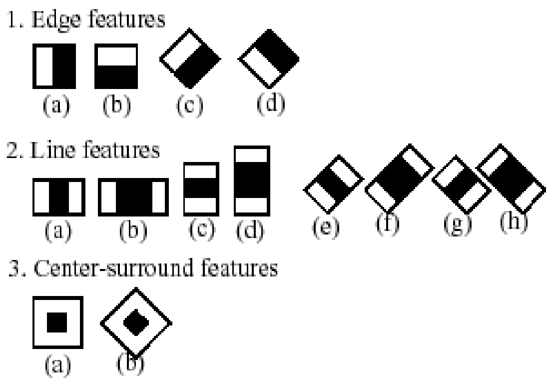
SVM

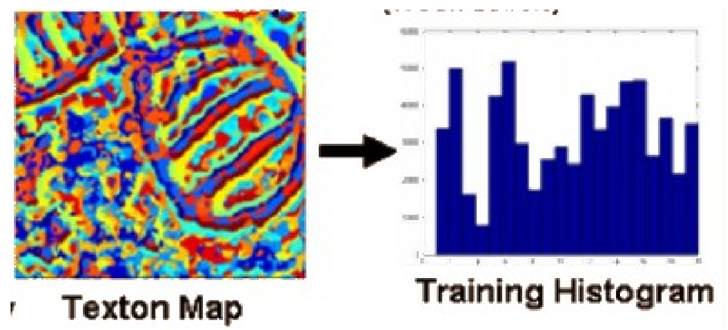
- Goal: find a hyperplane that divides the points having $y_i = -1$ from those having $y_i = 1$
- Any hyperplane can be written as the set of points $\mathbf{x}$ satisfying

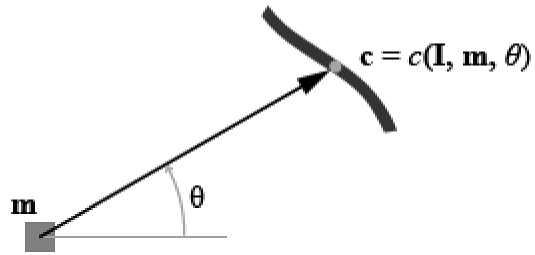
$$\mathbf{w}^\top \mathbf{x_i} + \mathbf{b} = 0$$

- Positive class: $\mathbf{w}^\top \mathbf{x_i} + \mathbf{b} \geq 1$
- Negative class: $\mathbf{w}^\top \mathbf{x_i} + \mathbf{b} \leq -1$

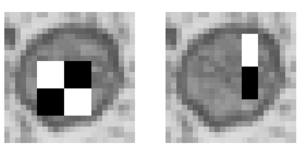
Large Choice of Features

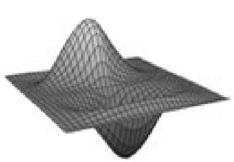


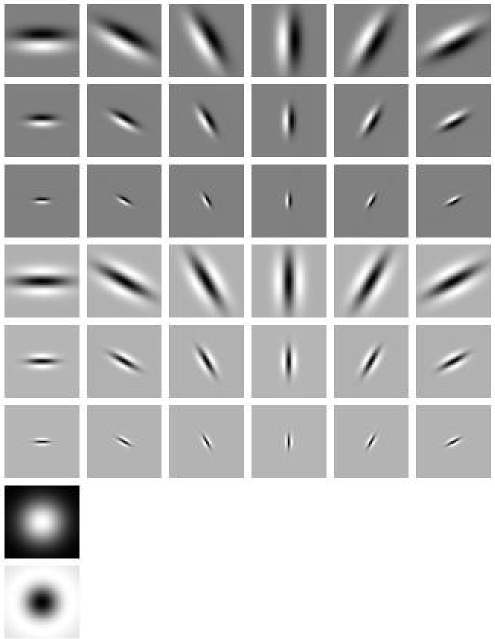


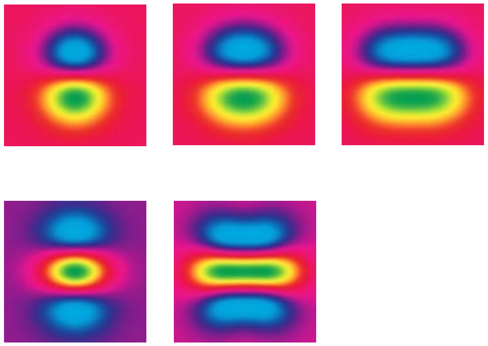


The characteristic function returns the nearest edge point **c** *in direction θ given location* **m**.

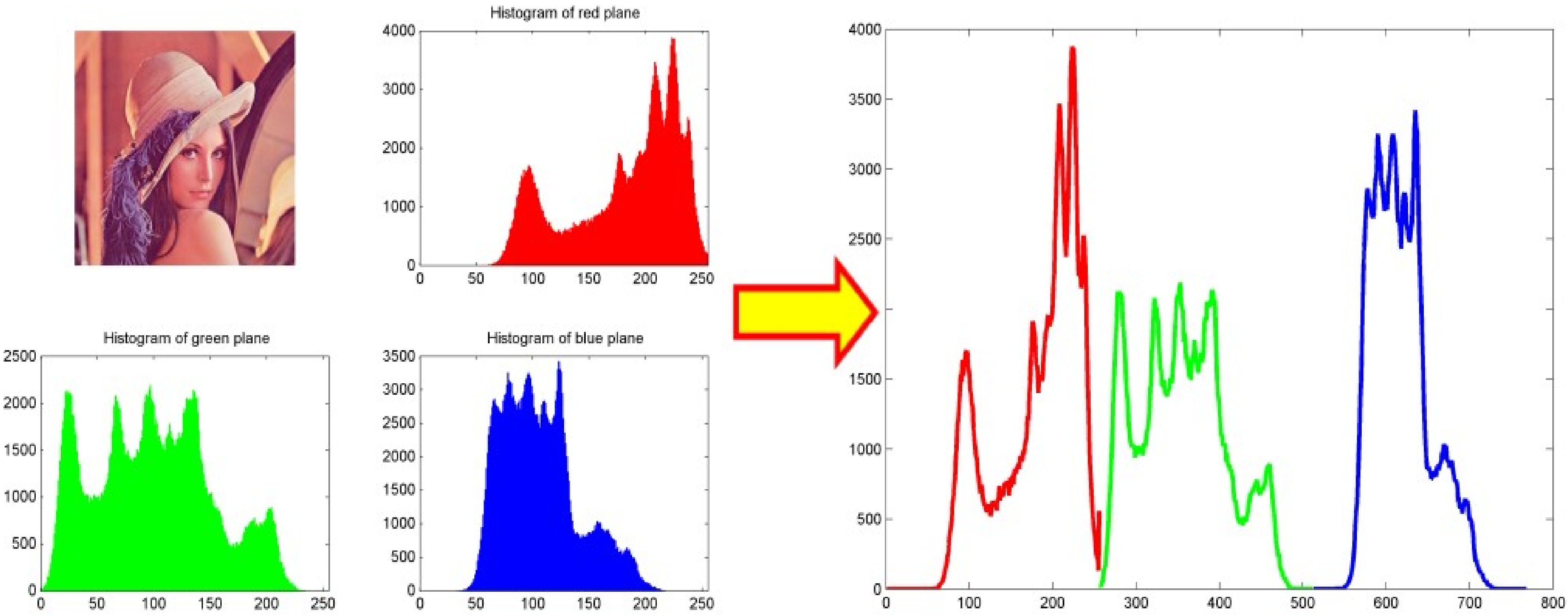






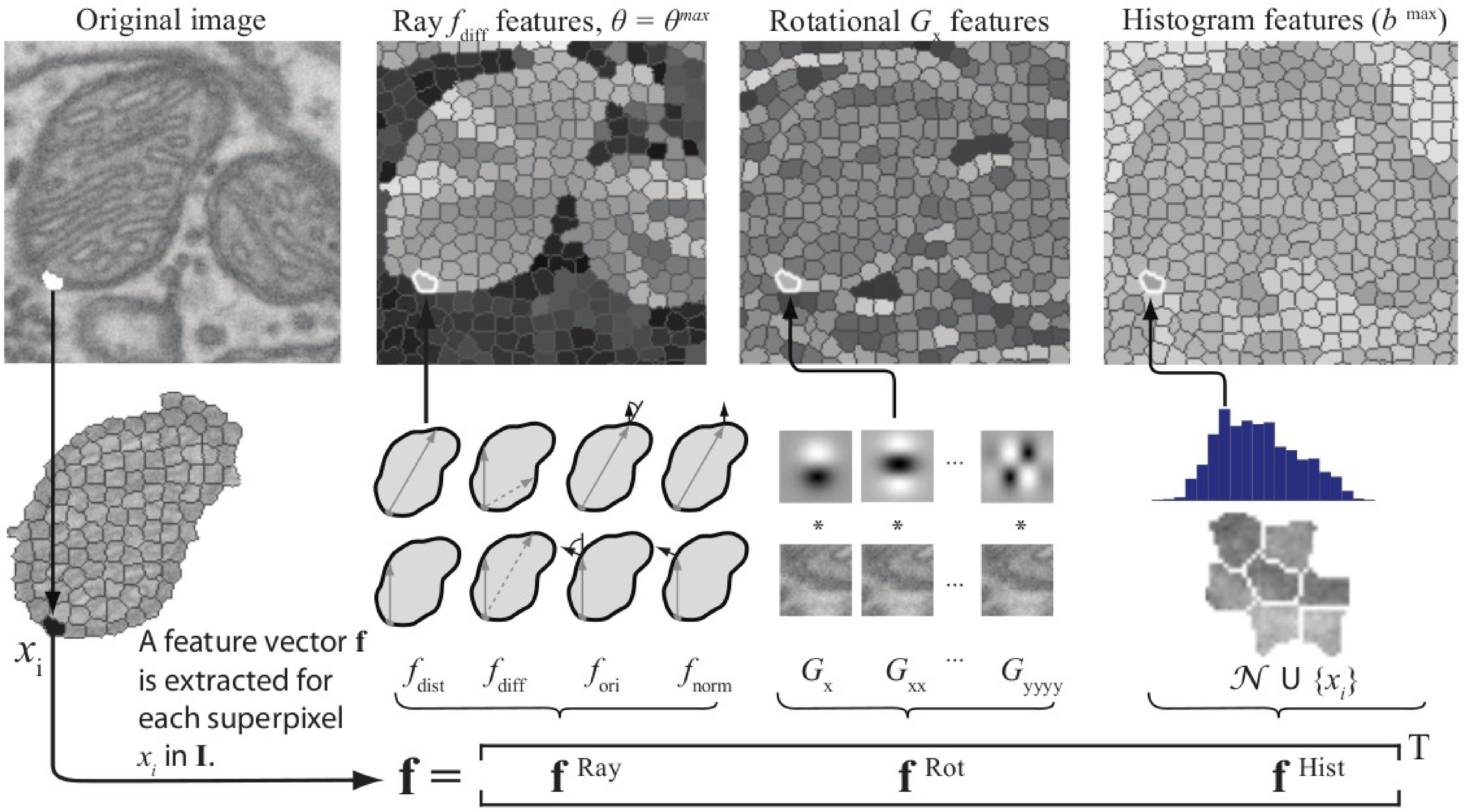


Histogram



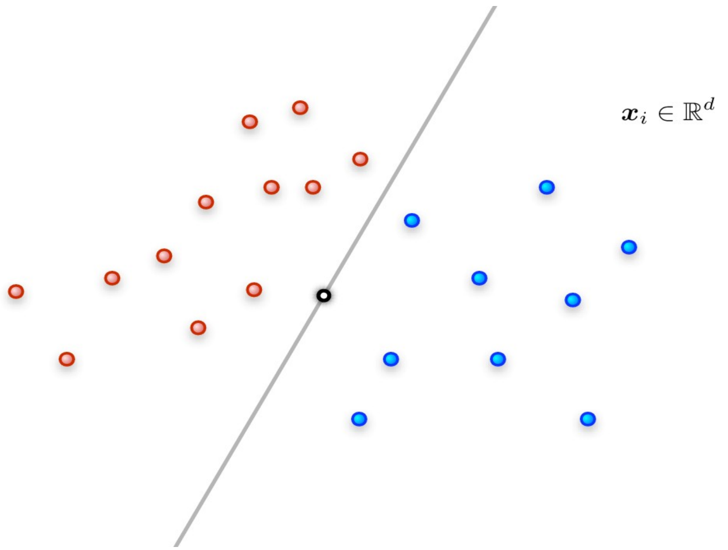
http://www.intechopen.com/source/html/39030/media/image2.png

Combining features



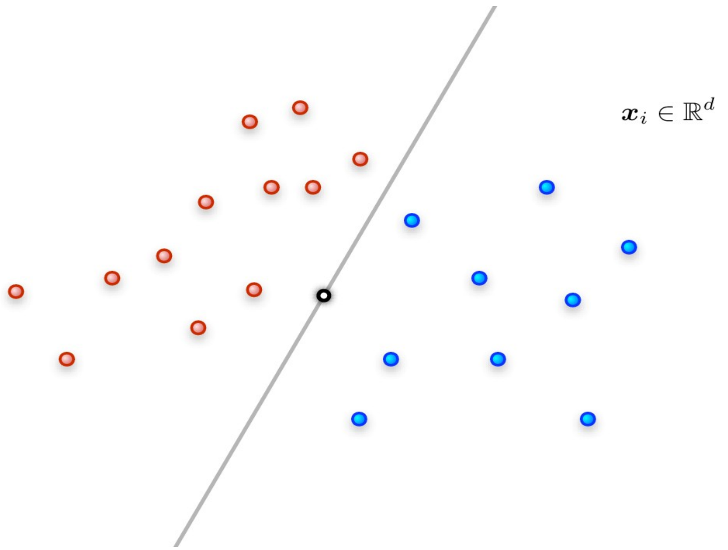
SVM

- Once the features are extracted, SVM finds the optimal classifier that separates them
- This amount to solving a quadratic program
- **Good news:** SVM can (almost) be used as a "black-box"



Perceptron

- Finds a hyperplane that separates the two sets
- Does not try to optimize the margin (unlike SVM).
- **Good news:** quite easy to implement (see Matlab exercise)

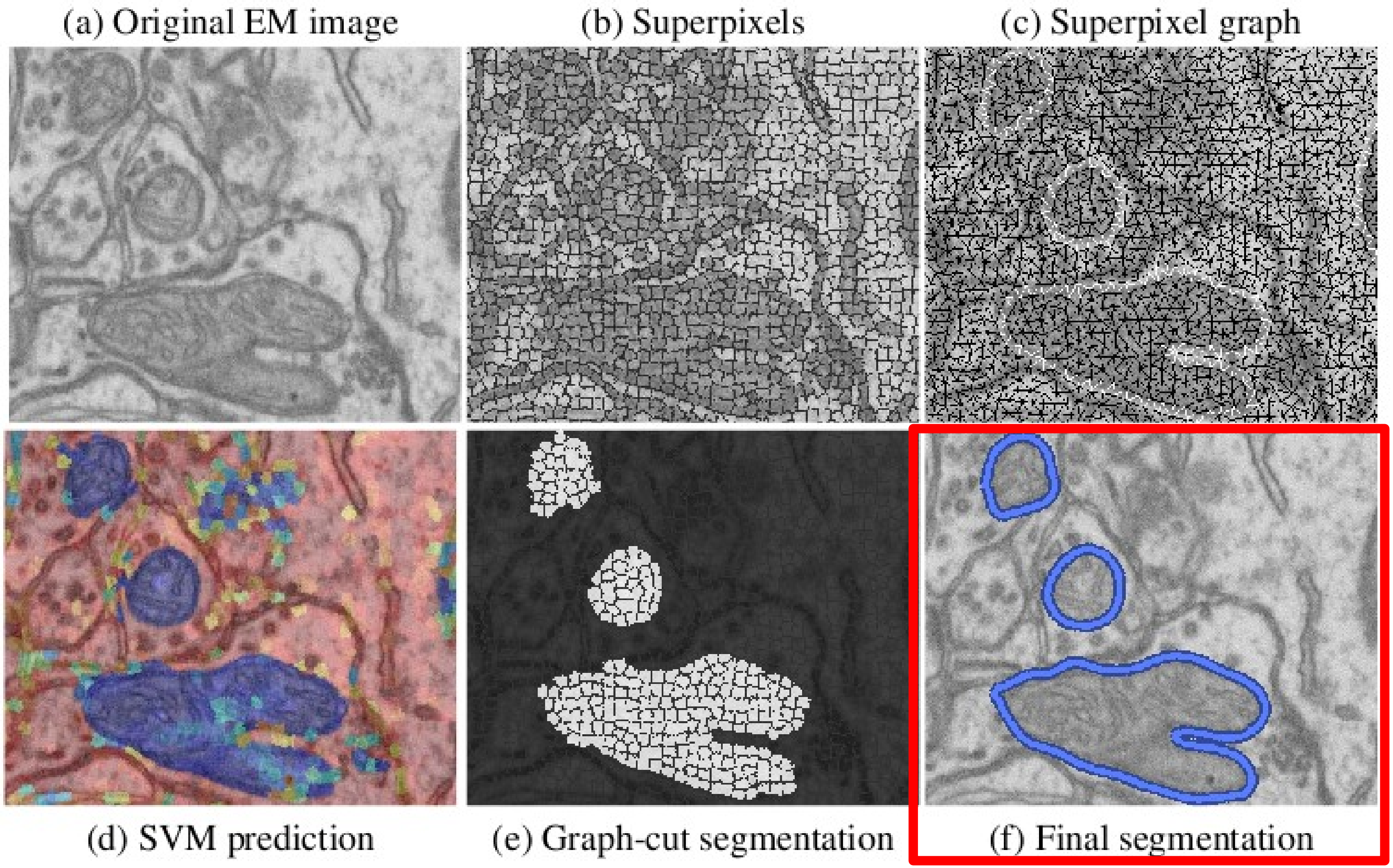


ETH zürich

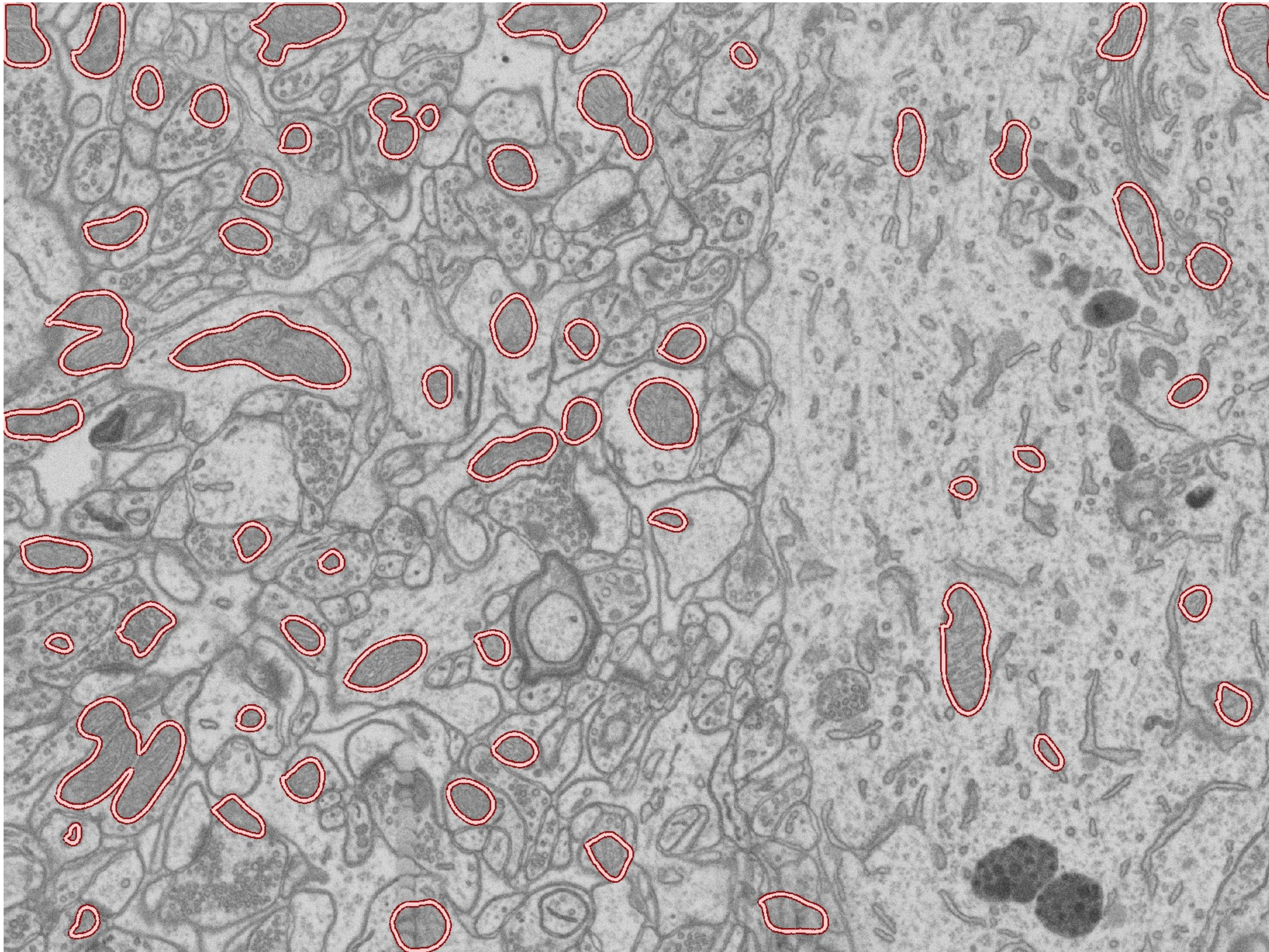
Other classifiers

- Boosting
- Random trees
- Gaussian processes
- Kernel SVM (only consider linear SVM in this presentation)
- …

Approach

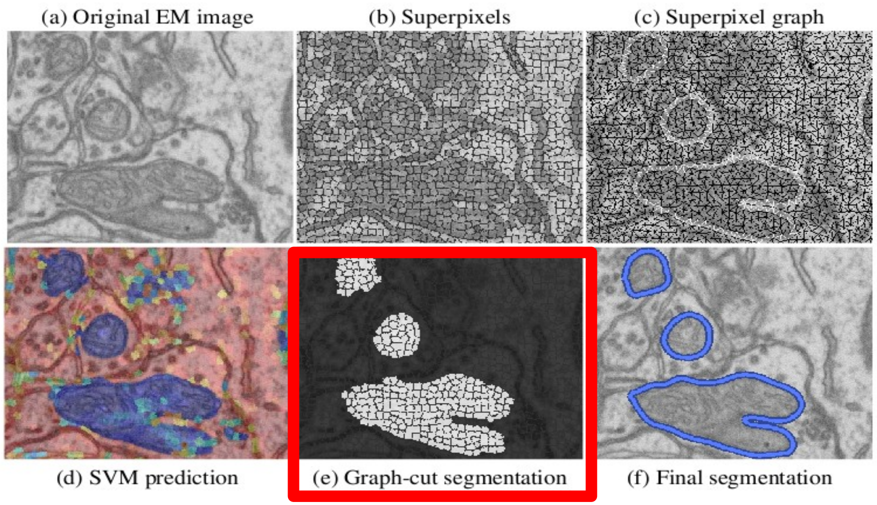


ETH zürich



Summary part 1

- (Extract superpixels)
- Compute features for each pixel/superpixel
- Use an SVM to label each pixel/superpixel
- Coming next: Use graph-cut to smooth the results



2. Structured Prediction for Image Segmentation

Keywords: Markov Random Field, Conditional Random Field, Graph-cut, ...

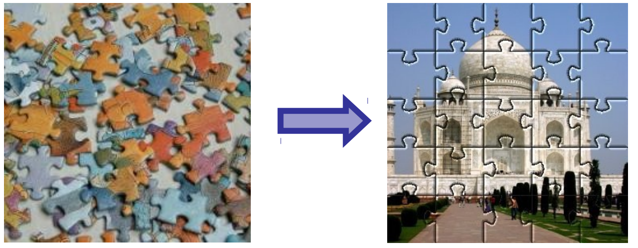
Structured Prediction

- **Non-structured output**

  $$f : X \to R$$

  - inputs X can be any kind of objects
  - output y is a real number

  $$y = \{-1, +1\}$$



- **Prediction of complex outputs**

  $$f : X \to Y$$

  - Structured output y is complex (images, text, audio...)
  - Ad hoc definition of structured data: data that consists of several parts, and not only the parts themselves contain information, but also the way in which the parts belong together

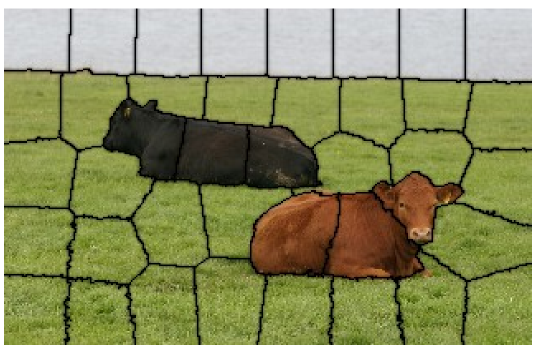
ETHzürich

Slide courtesy: Christoph Lampert

Structured Prediction for Images

$$X$$

$$(x_1, \cdots, x_i, \cdots, x_n)$$

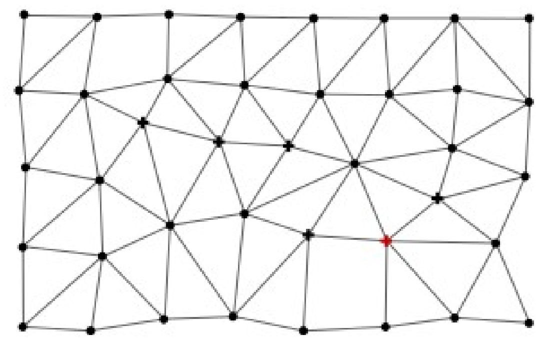


$$x_i \in \mathbb{R}^F$$

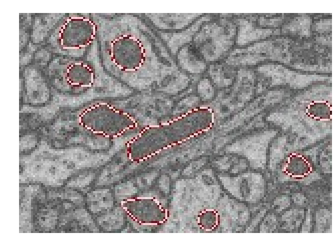
Histograms, Filter responses, ...

$$Y$$

$$(y_1, \cdots, y_i, \cdots, y_n)$$



$$y_i = \{-1, +1\}$$



$$y_i = \{1, \ldots 21\}$$

Original Ground truth

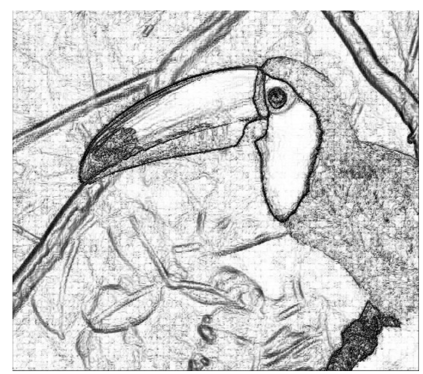
CRF for Image Segmentation

- Define Energy as a function of X and Y, parametrized by w
- **Goal:** Find minimum of this function that corresponds to the best solution

$$E_{\mathbf{w}}(X,Y) = \sum_{i \in \mathcal{V}} D_{\mathbf{w}}(y_i; x_i) + \sum_{(i,j) \in \mathcal{E}} V_{\mathbf{w}}(y_i, y_j)$$

Data (**D**)

Unary likelihood



Pair-wise Terms

MAP Solution

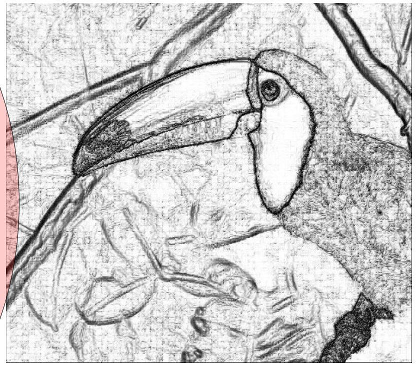
Boykov and Jolly [ICCV 2001], Blake et al. [ECCV 2004]
Slide courtesy : Pushmeet Kohli

CRF for Image Segmentation

$$E_{\mathbf{w}}(X,Y) = \sum_{i \in \mathcal{V}} D_{\mathbf{w}}(y_i; x_i) + \sum_{(i,j) \in \mathcal{E}} V_{\mathbf{w}}(y_i, y_j)$$

Data (**D**)

Unary likelihood



Pair-wise Terms

MAP Solution

Boykov and Jolly [ICCV 2001], Blake et al. [ECCV 2004]
Slide courtesy : Pushmeet Kohli

CRF for Image Segmentation

$$E_{\mathbf{w}}(X,Y) = \sum_{i \in \mathcal{V}} D_{\mathbf{w}}(y_i; x_i) + \cancel{\sum_{(i,j) \in \mathcal{E}} V_{\mathbf{w}}(y_i, y_j)}$$

$$D_{\mathbf{w}}(y_i, x_i) = \mathbf{w}_{\mathbf{y_i}}^{\top} x_i$$

Data (**D**)

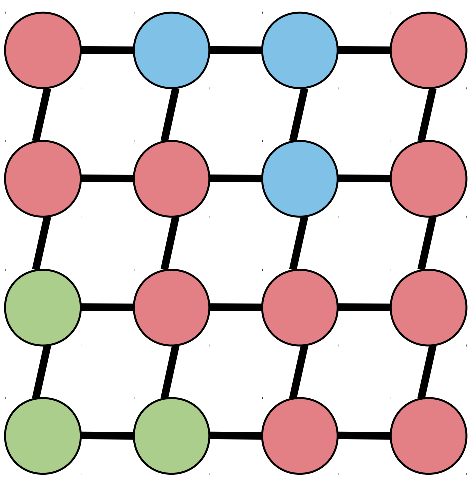
Unary likelihood

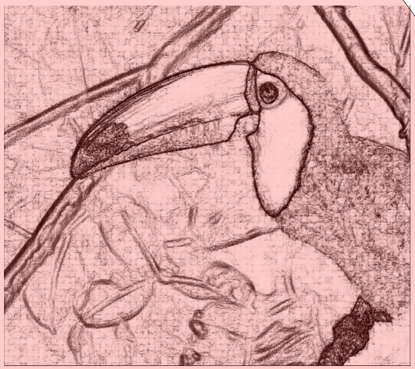
The unary likelihood term alone would be equivalent to the SVM formulation.

Boykov and Jolly [ICCV 2001], Blake et al. [ECCV 2004]
Slide courtesy : Pushmeet Kohli

CRF for Image Segmentation

$$E_{\mathbf{w}}(X,Y) = \sum_{i \in \mathcal{V}} D_{\mathbf{w}}(y_i; x_i) + \sum_{(i,j) \in \mathcal{E}} V_{\mathbf{w}}(y_i, y_j)$$





Pair-wise Terms

Favors the same label for neighboring nodes.

CRF for Image Segmentation

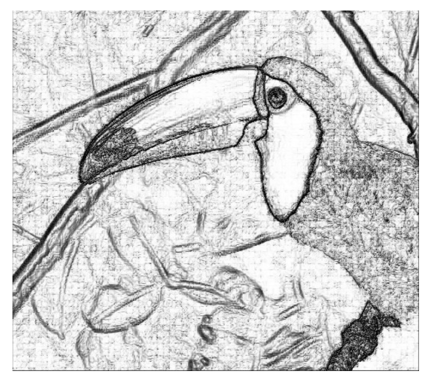
$$E_{\mathbf{w}}(X,Y) = \sum_{i \in \mathcal{V}} D_{\mathbf{w}}(y_i; x_i) + \sum_{(i,j) \in \mathcal{E}} V_{\mathbf{w}}(y_i, y_j)$$

Maximum-a-posteriori (MAP) solution :

$$Y^* = \arg\min_{Y \in \mathcal{Y}} E_w(X,Y)$$

Data (**D**)

Unary likelihood

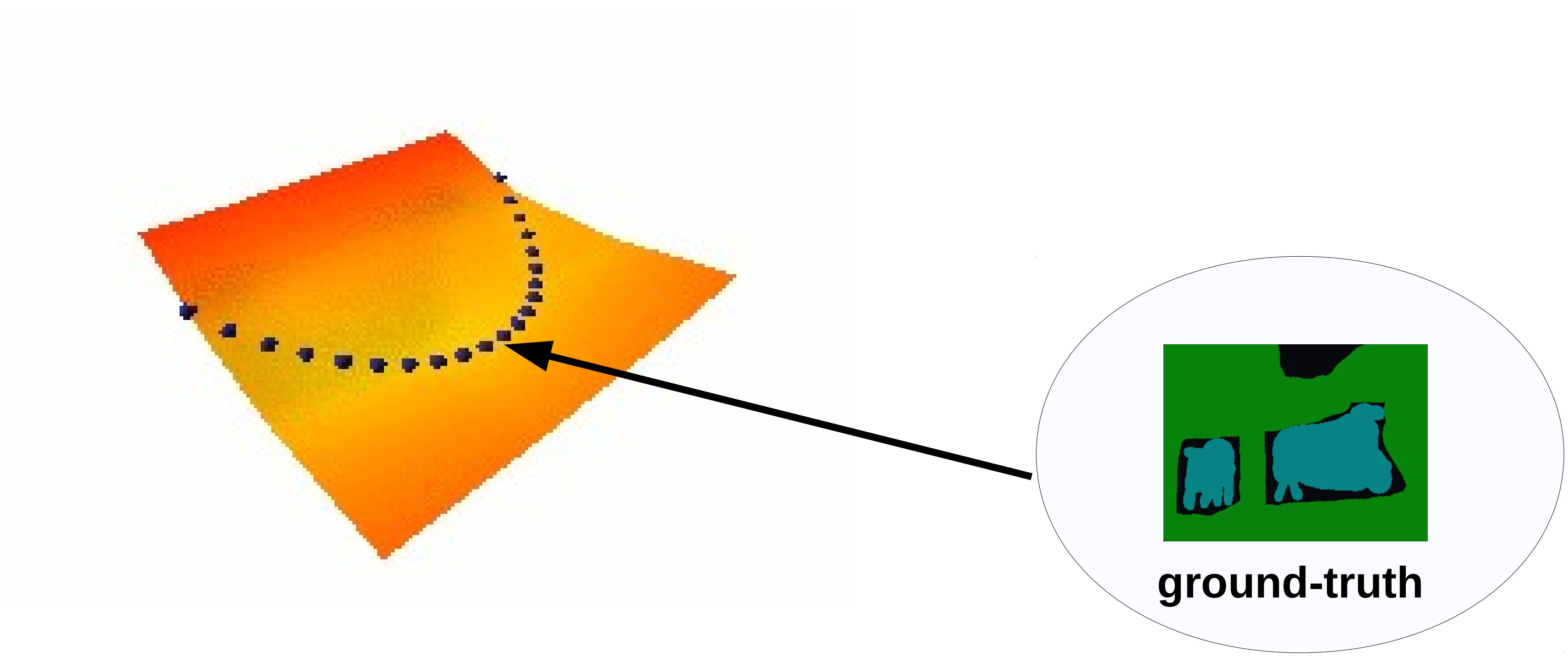


Pair-wise Terms

MAP Solution

Boykov and Jolly [ICCV 2001], Blake et al. [ECCV 2004]
Slide courtesy : Pushmeet Kohli

Energy Based Landscape

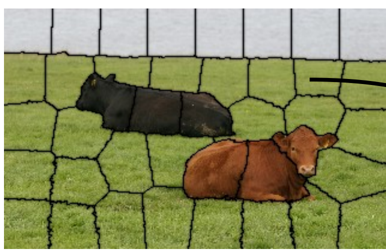


See http://www.cs.nyu.edu/~yann/research/ebm/

Feature map

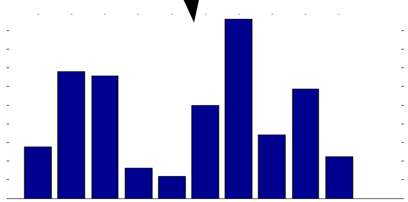
- Energy function is parametrized by vector **w**

$$E_{\mathbf{w}}(X,Y) = \sum_{i \in \mathcal{V}} D_{\mathbf{w}}(y_i; x_i) + \sum_{i,j \in \mathcal{E}} V_{\mathbf{w}}(y_i, y_j) = \mathbf{w}^T \psi(X,Y)$$



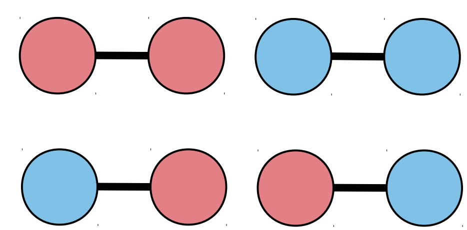
$$\mathbf{w} = ((\mathbf{w}^D)^T, (\mathbf{w}^V)^T)^T$$

$(\mathbf{w}^D)^T$



+

$\mathbf{w}^P$	-1	1
-1	?	?
1	?	?



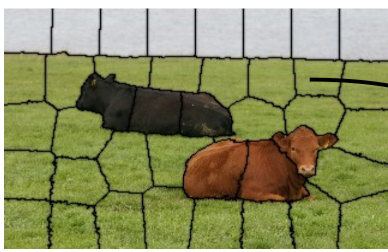
$$D(y_i) = (\mathbf{w}^D)^T x_i$$

$$V(y_i, y_j) = \mathbf{w}^V(y_i, y_j)$$

Feature map

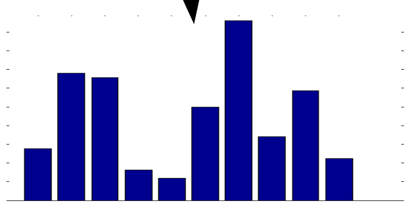
- Energy function is parametrized by vector **w**

$$E_{\mathbf{w}}(X,Y) = \sum_{i\in\mathcal{V}} D_{\mathbf{w}}(y_i;x_i) + \sum_{i,j\in\mathcal{E}} V_{\mathbf{w}}(y_i,y_j) = \mathbf{w}^T\psi(X,Y)$$



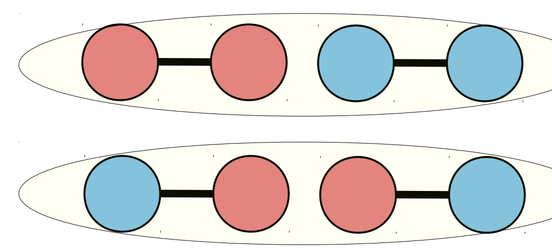
$$\mathbf{w} = ((\mathbf{w}^D)^T,(\mathbf{w}^V)^T)^T$$

$(\mathbf{w}^D)^T$



+

$\mathbf{w}^P$	-1	1
-1	0	1
1	1	0



$$D(y_i) = (\mathbf{w}^D)^T x_i$$

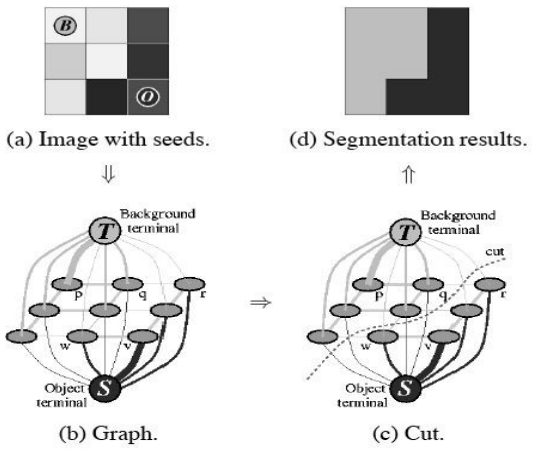
$$V(y_i,y_j) = \mathbf{w}^V(y_i,y_j)$$

Energy Minimization

- MAP inference for discrete graphical models:

$$Y^* = \arg\min_{Y \in \mathcal{Y}} E_w(X, Y)$$

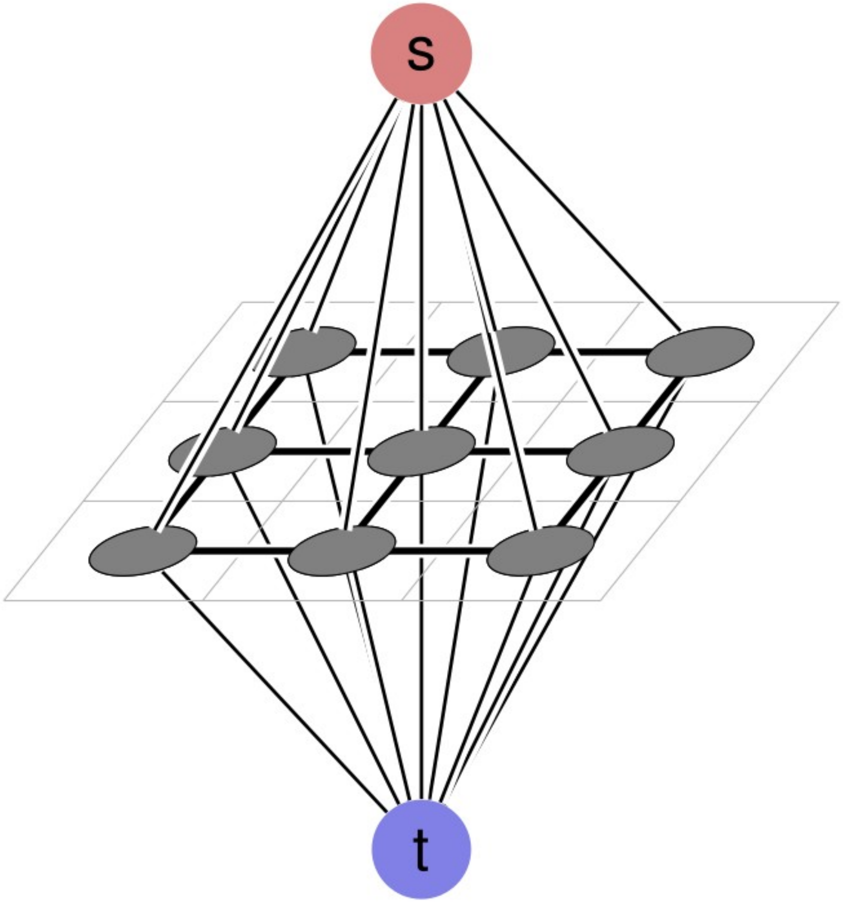
  - Dynamic programming
    - Exact on non loopy graphs
  - **Graph-cuts (Boykov, 2001)**
    - Optimal solution if energy function is submodular
  - Belief propagation (Pearl, 1982)
    - No theoretical guarantees on loopy graphs but seems to work well in practice.
  - Mean field (root in statistical physics)
  - ...



ETH zürich

Graph-cuts

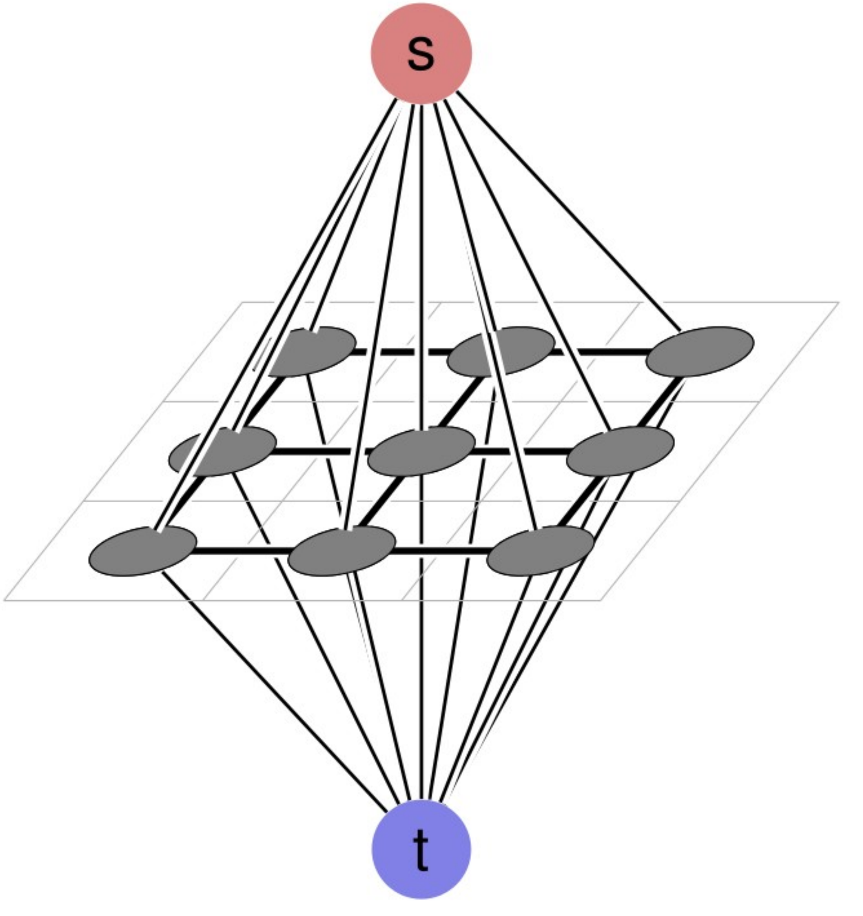
- Consider a directed graph with **n + 2** nodes, with
  - a “source” node s,
  - a “sink” node t,
  - and n (super)pixels.



Graph-cuts

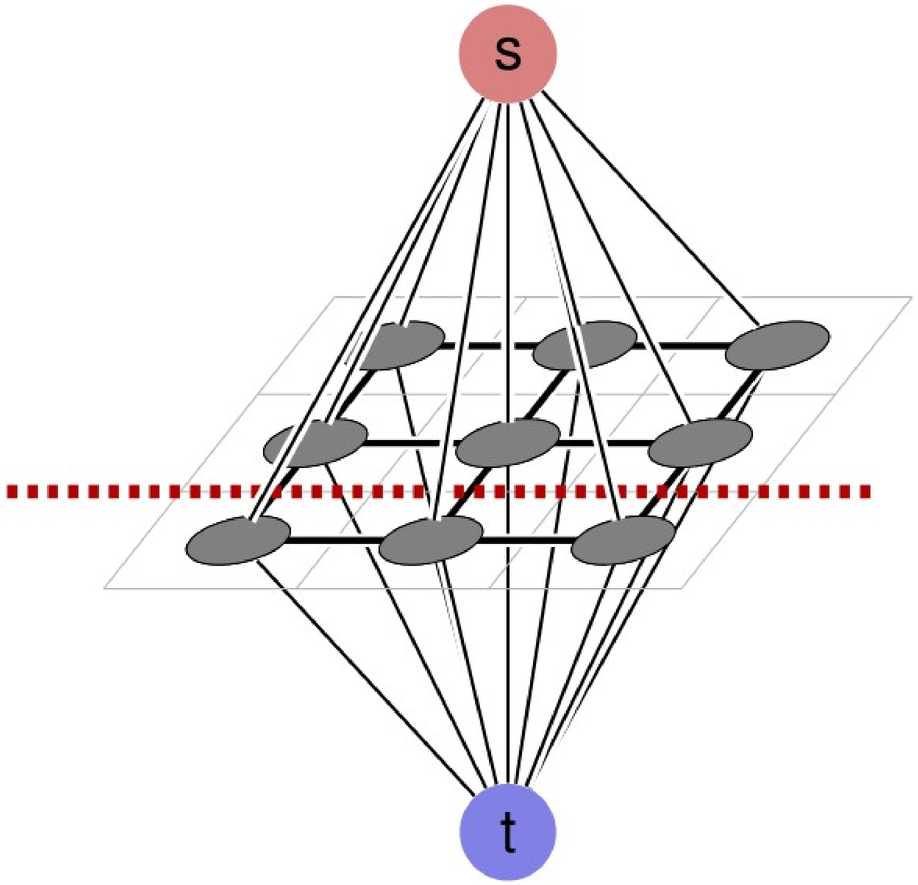
- Associate a cost to each node and each edge
- These costs somehow relate to the unary and pairwise terms in our energy function

$$E_{\mathbf{w}}(X,Y) = \sum_{i \in \mathcal{V}} D_{\mathbf{w}}(y_i; x_i) + \sum_{(i,j) \in \mathcal{E}} V_{\mathbf{w}}(y_i, y_j)$$



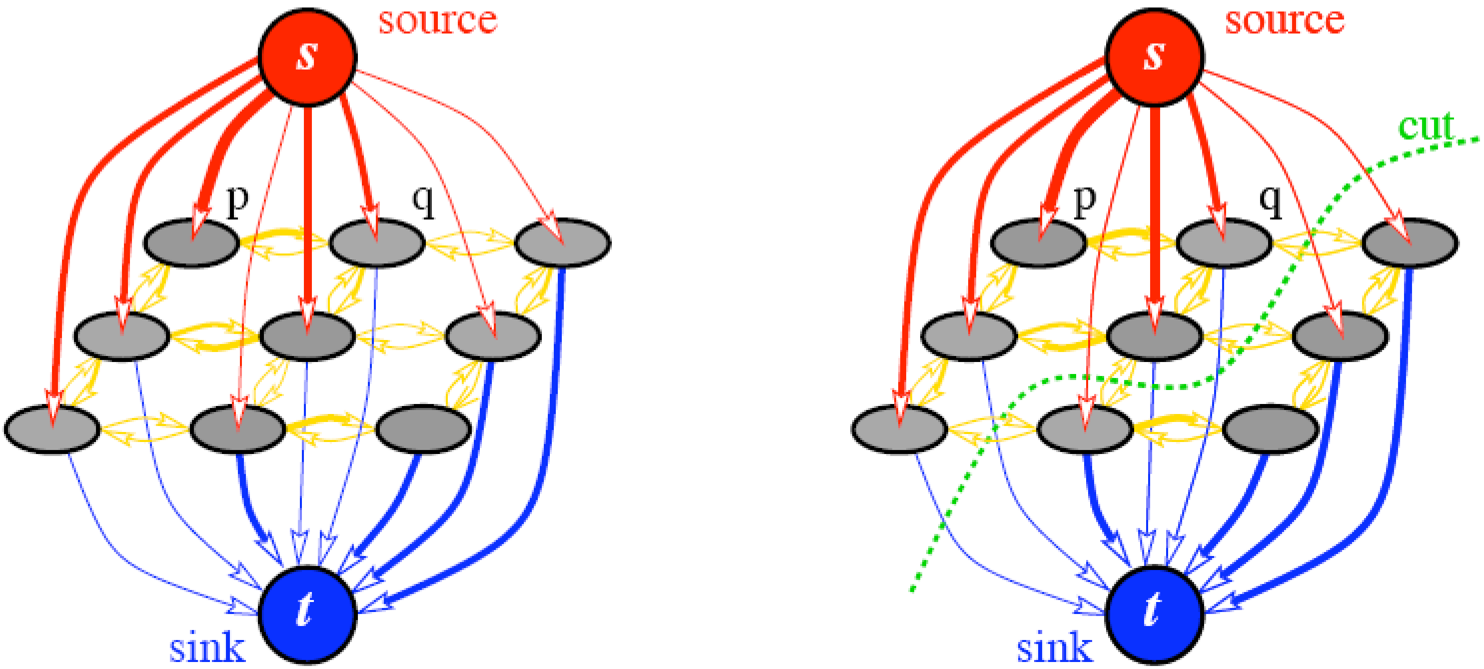
Graph-cuts

- **Goal:** find a partitioning that minimizes the sum of costs



Graph-cuts

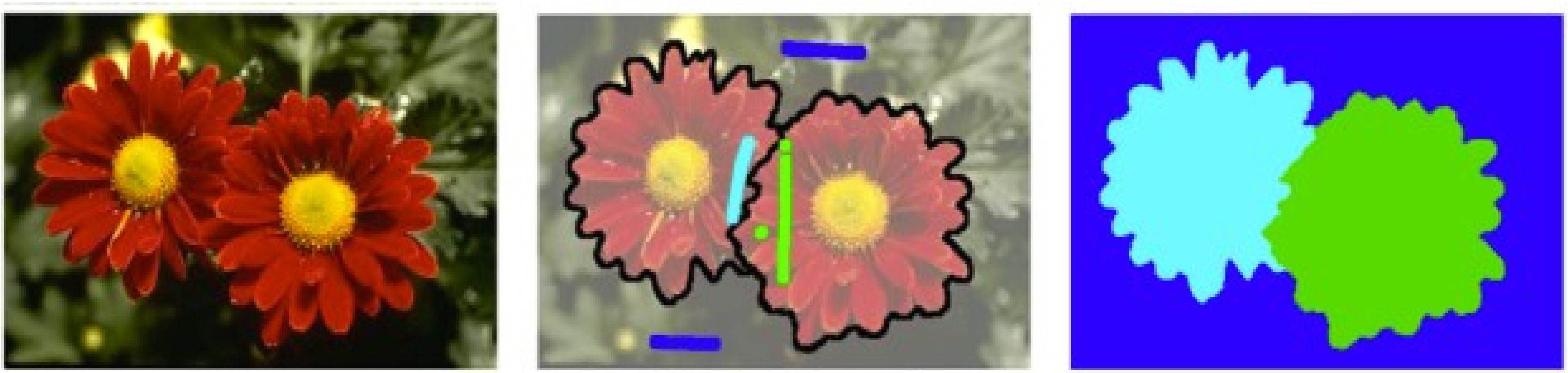
- **How?** Mincut-maxflow algorithm



ETH zürich

Graph-cuts

- Graph-cuts allows a user to enter seeds (e.g. via a mouse operated brush)



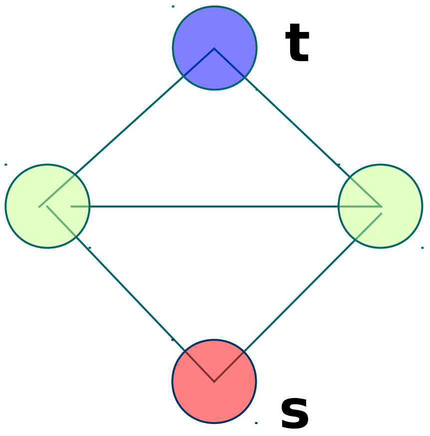
Graph-cuts

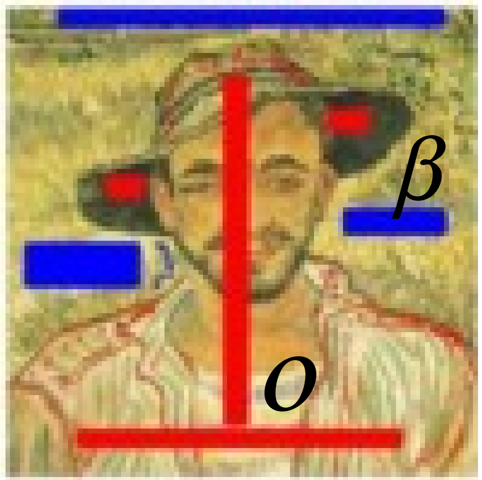
- How to set the weights?

edge	weight (cost)	for
$\{p,q\}$	$B_{\{p,q\}}$	$\{p,q\} \in \mathcal{N}$
$\{p,S\}$	$\lambda \cdot R_p(\text{"bkg"})$	$p \in \mathcal{P},\ p \notin \mathcal{O} \cup \mathcal{B}$
	K	$p \in \mathcal{O}$
	0	$p \in \mathcal{B}$
$\{p,T\}$	$\lambda \cdot R_p(\text{"obj"})$	$p \in \mathcal{P},\ p \notin \mathcal{O} \cup \mathcal{B}$
	0	$p \in \mathcal{O}$
	K	$p \in \mathcal{B}$

where

$$K = 1 + \max_{p \in \mathcal{P}} \sum_{q:\ \{p,q\} \in \mathcal{N}} B_{\{p,q\}}.$$

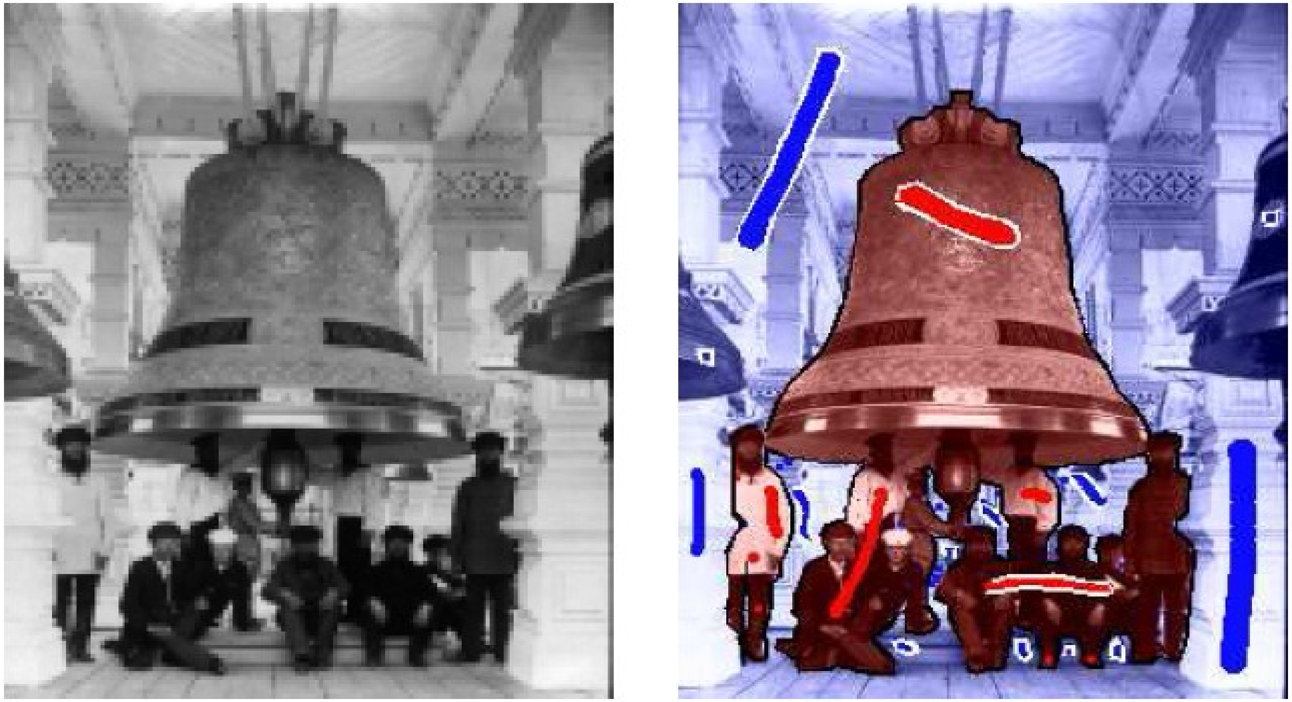




Boykov and Jolly [ICCV 2001]

Graph-cuts

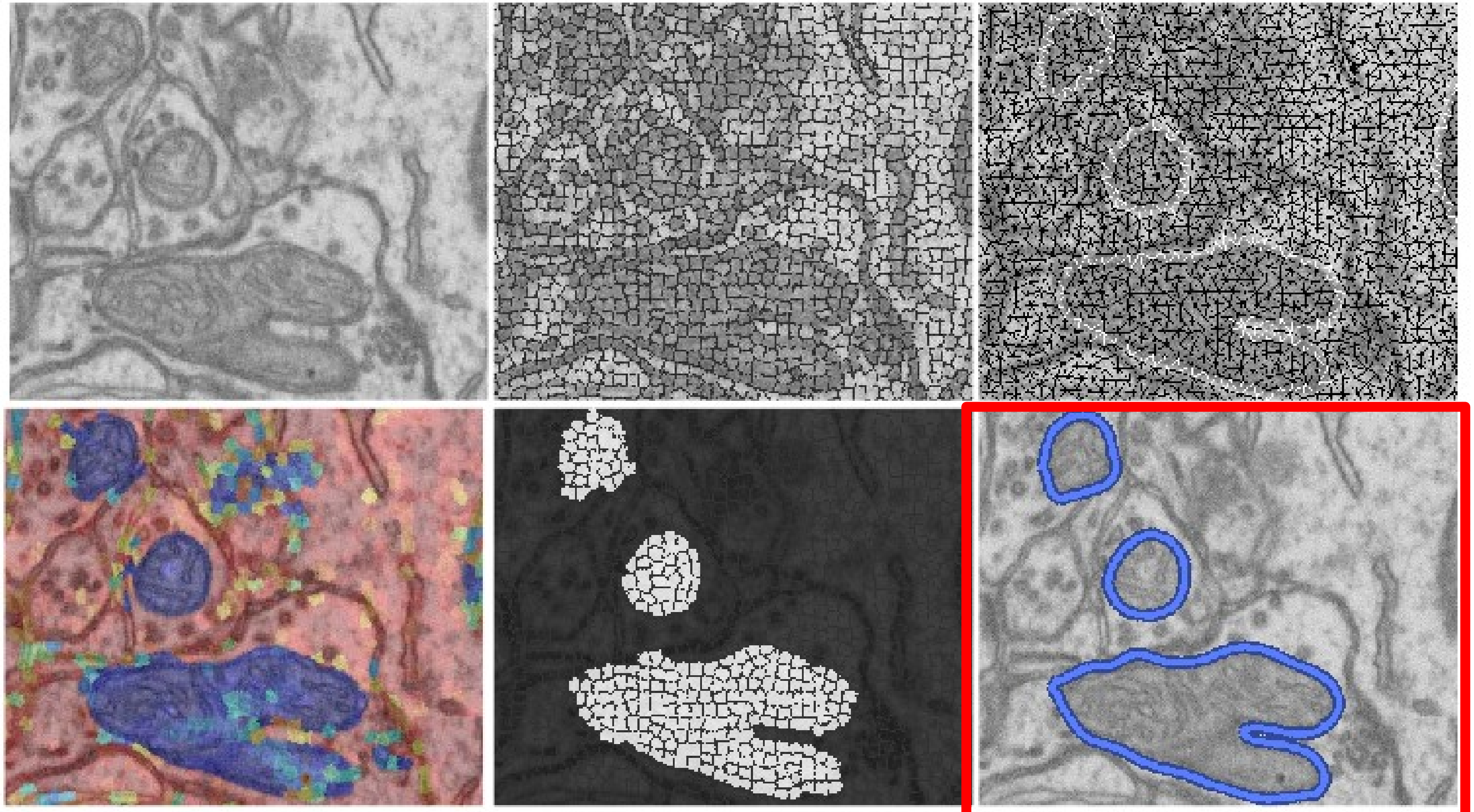
Photo Editing



(a) Bell Photo (b) Bell Segmentation

Approach

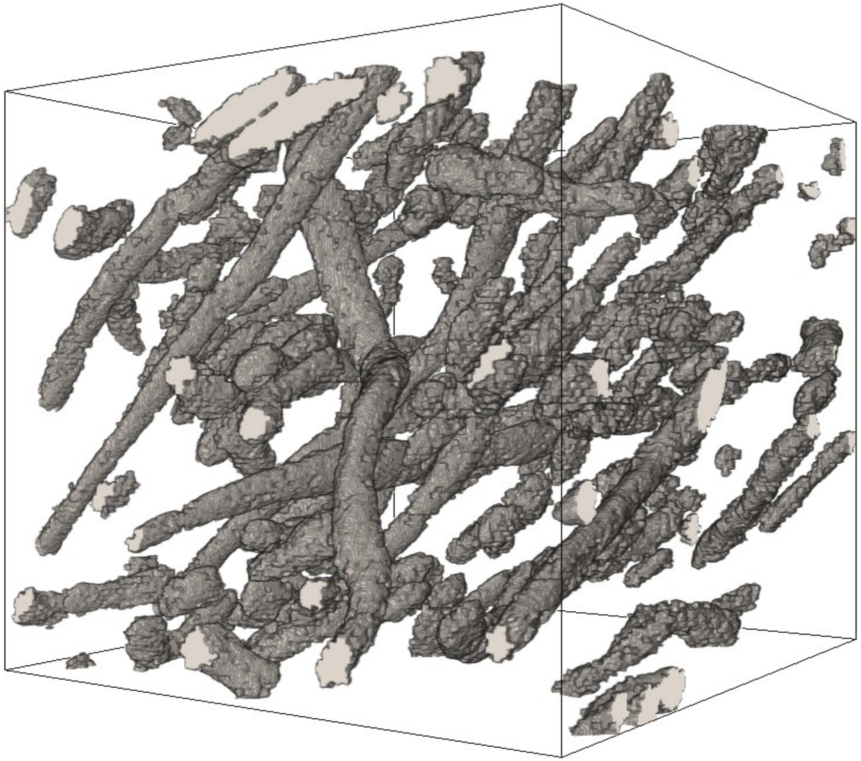
(a) Original EM image (b) Superpixels (c) Superpixel graph

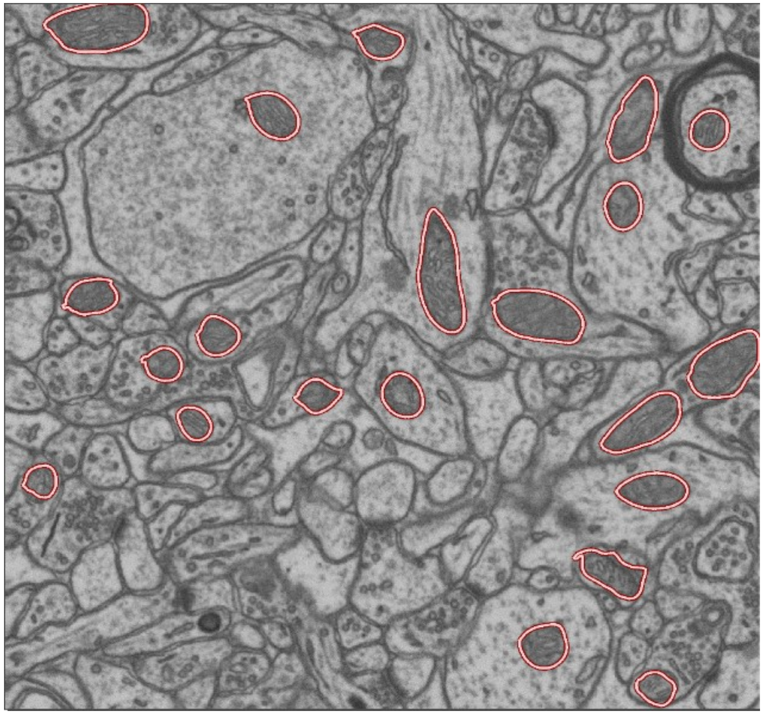


(d) SVM prediction (e) Graph-cut segmentation (f) Final segmentation

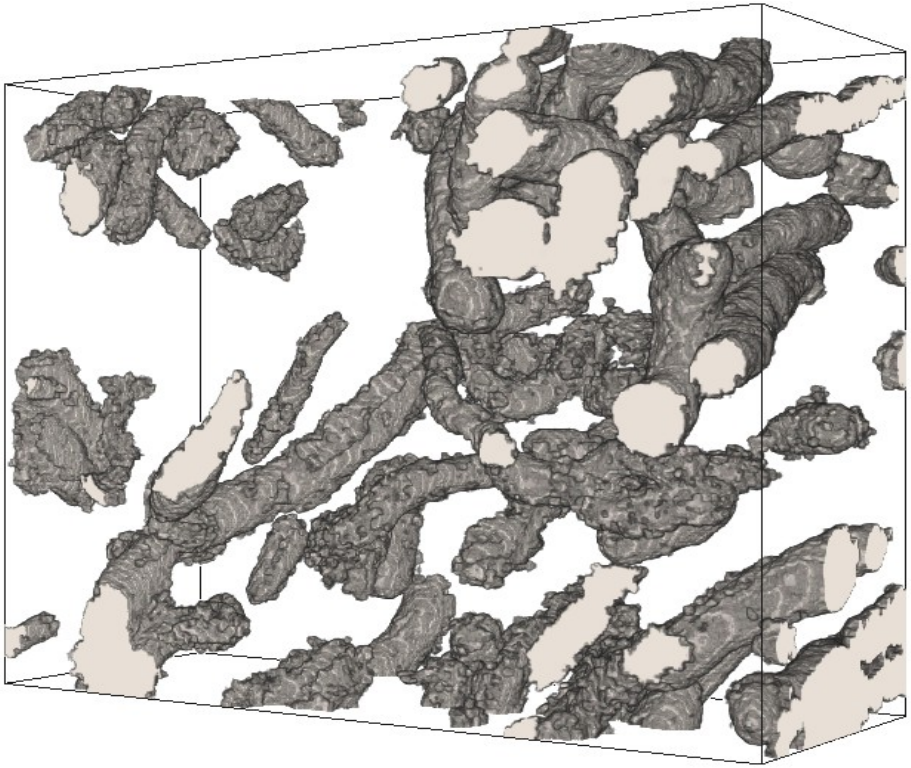
ETH zürich

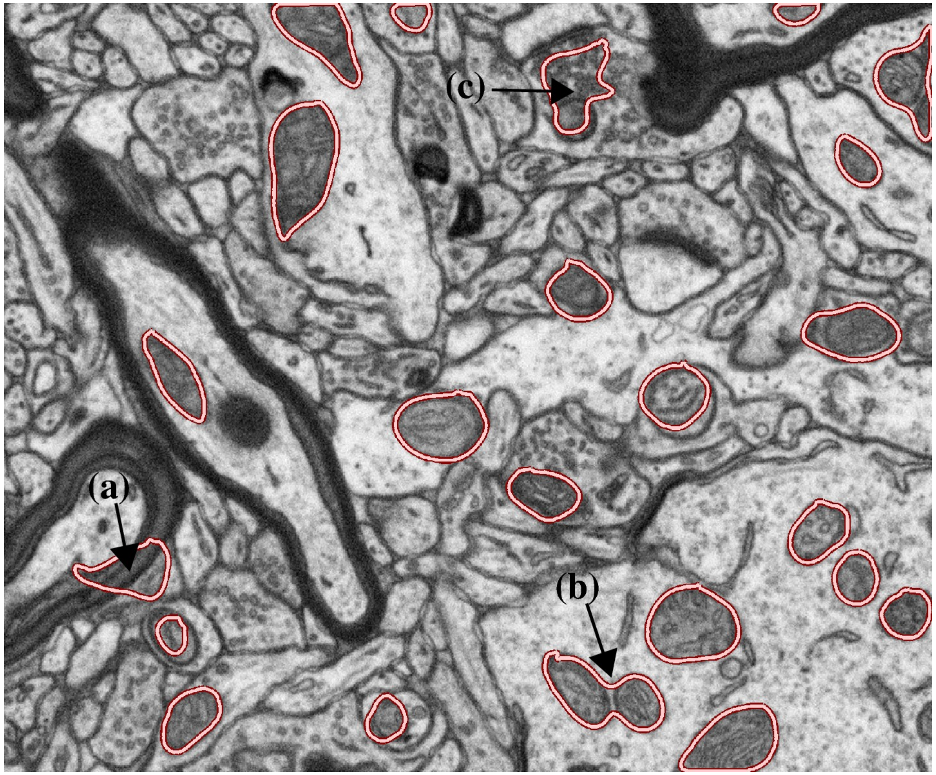
Results - Hippocampus





Results - Striatum

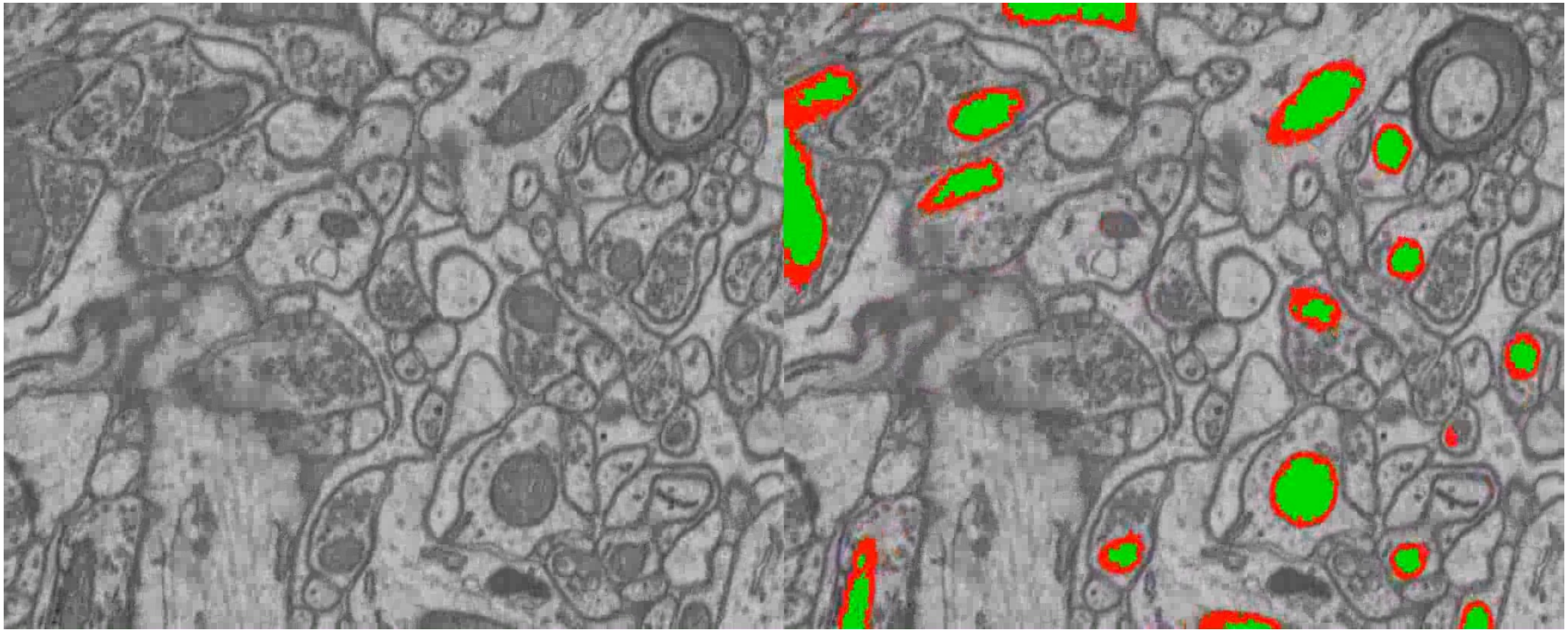




ETH zürich

Results - Hippocampus

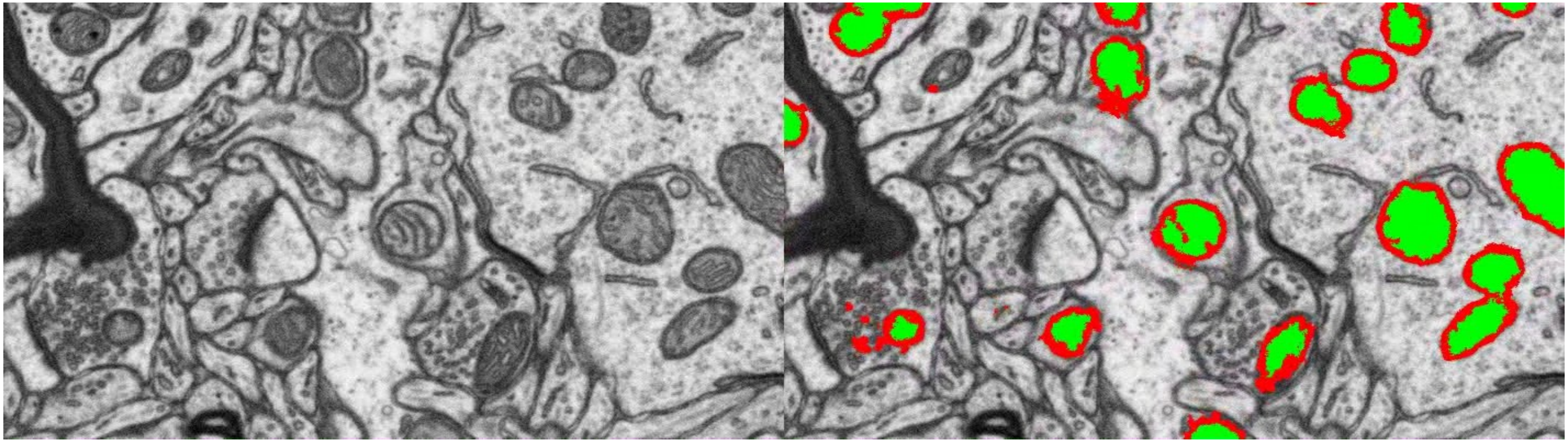
hippocampus.avi



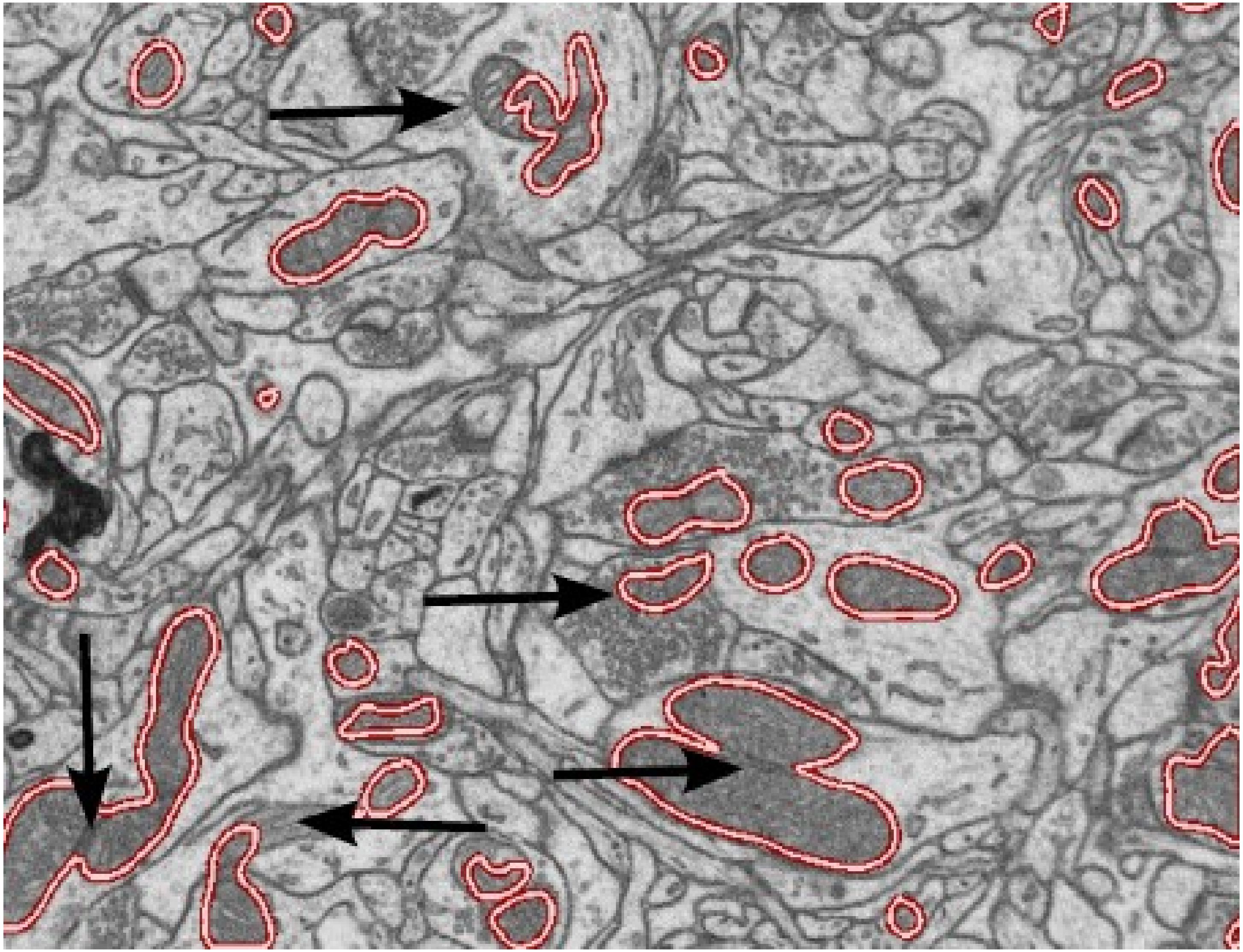
ETH zürich

Results - Striatum

striatum.avi



Remaining Errors



Questions

Credits: Martin Jaggi, Kevin Mader