Extensions and applications of Boosting

Basic Algorithm and Core Theory

- introduction to AdaBoost
- analysis of training error
- analysis of test error
  and the margins theory
- experiments and applications

A Formal Description of Boosting

- given training set $(x_1, y_1), \ldots, (x_m, y_m)$
- $y_i \in \{-1, +1\}$ correct label of instance $x_i \in X$
- for $t = 1, \ldots, T$:
  - construct distribution D_t on $\{1, \ldots, m\}$
  - find weak classifier ("rule of thumb")

$$h_t : X \to \{-1, +1\}$$

with small error ϵ_t on D_t:

$$\epsilon_t = \Pr_{i \sim D_t}[h_t(x_i) \neq y_i]$$

- output final classifier H_{final}

AdaBoost

[Freund & Schapire 96]

- constructing D_t:
  - $D_1(i) = 1/m$
  - given D_t and h_t:

$$D_{t+1}(i) = \frac{D_t(i)}{Z_t} \times \begin{cases} e^{-\alpha_t} & \text{if } y_i = h_t(x_i) \\ e^{\alpha_t} & \text{if } y_i \neq h_t(x_i) \end{cases}$$

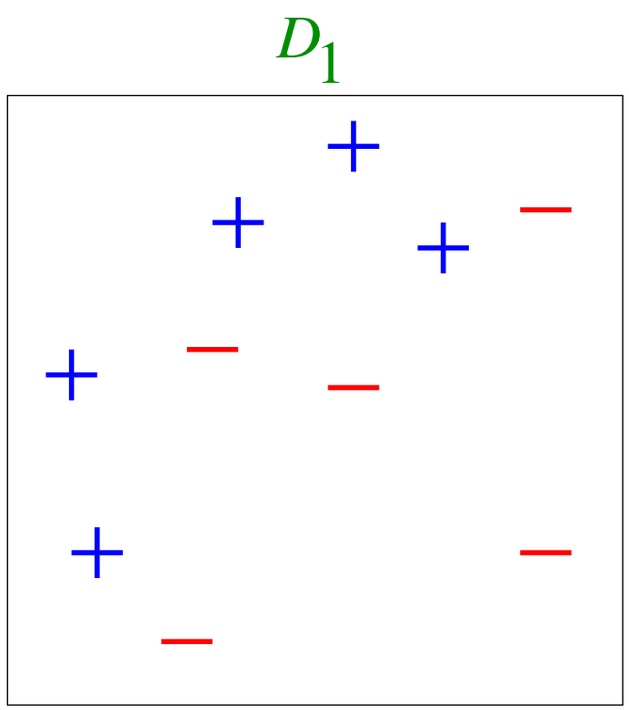
$$= \frac{D_t(i)}{Z_t} \exp(-\alpha_t \, y_i \, h_t(x_i))$$

where Z_t = normalization factor

$$\alpha_t = \frac{1}{2} \ln \left(\frac{1 - \epsilon_t}{\epsilon_t} \right) > 0$$

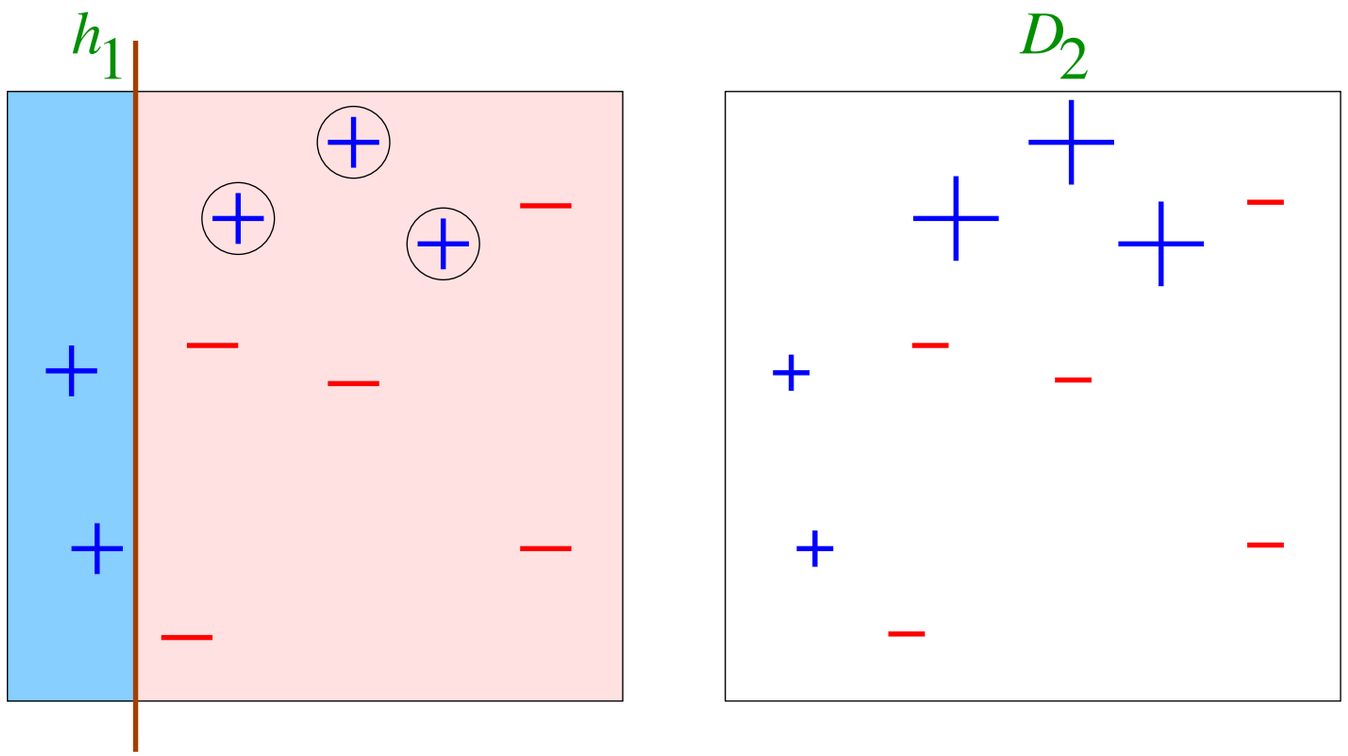
- final classifier:
  - $H_{\text{final}}(x) = \text{sign} \left(\sum_t \alpha_t h_t(x) \right)$

Toy Example



weak classifiers = vertical or horizontal half-planes

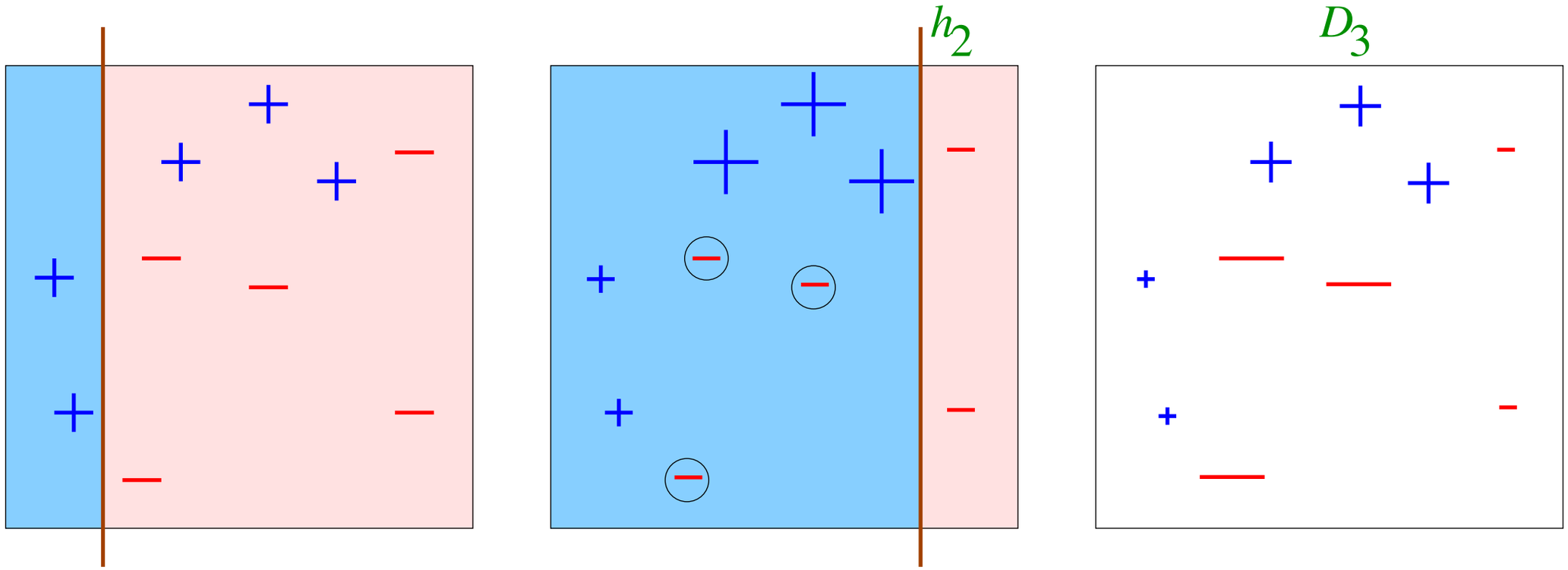
Round 1



$\varepsilon_1 = 0.30$

$\alpha_1 = 0.42$

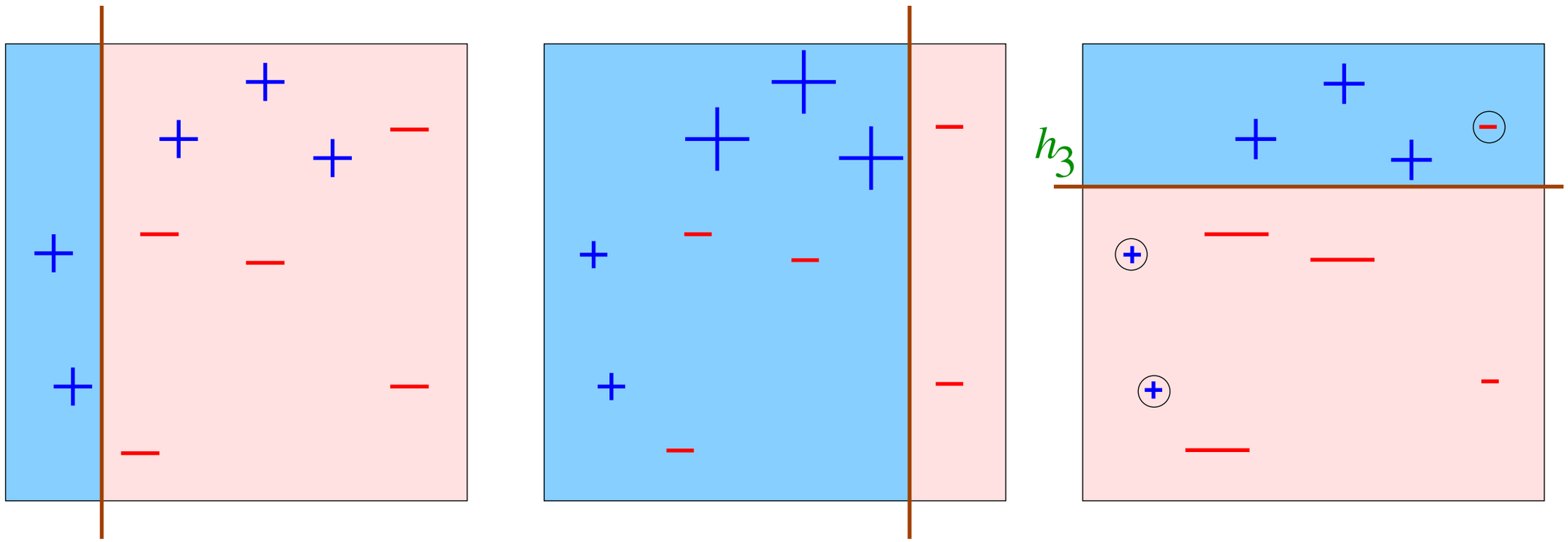
Round 2



$\varepsilon_2 = 0.21$

$\alpha_2 = 0.65$

Round 3

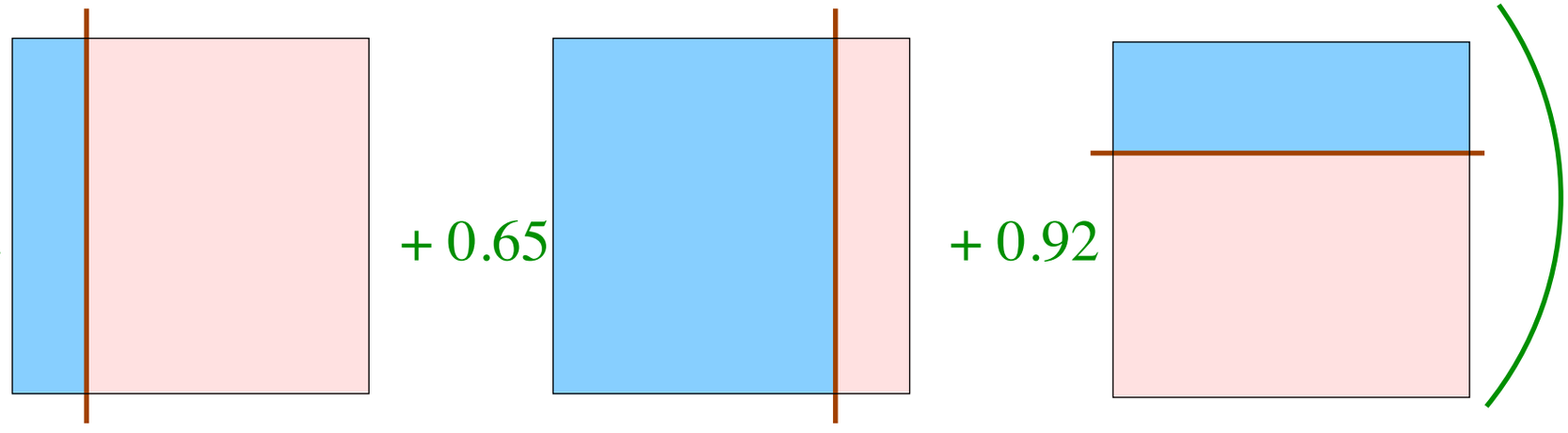


$\varepsilon_3 = 0.14$

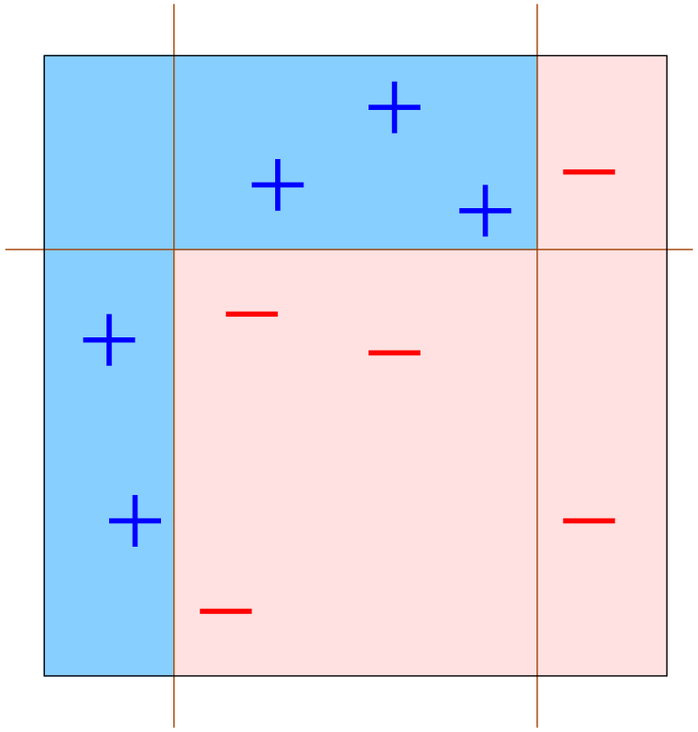
$\alpha_3 = 0.92$

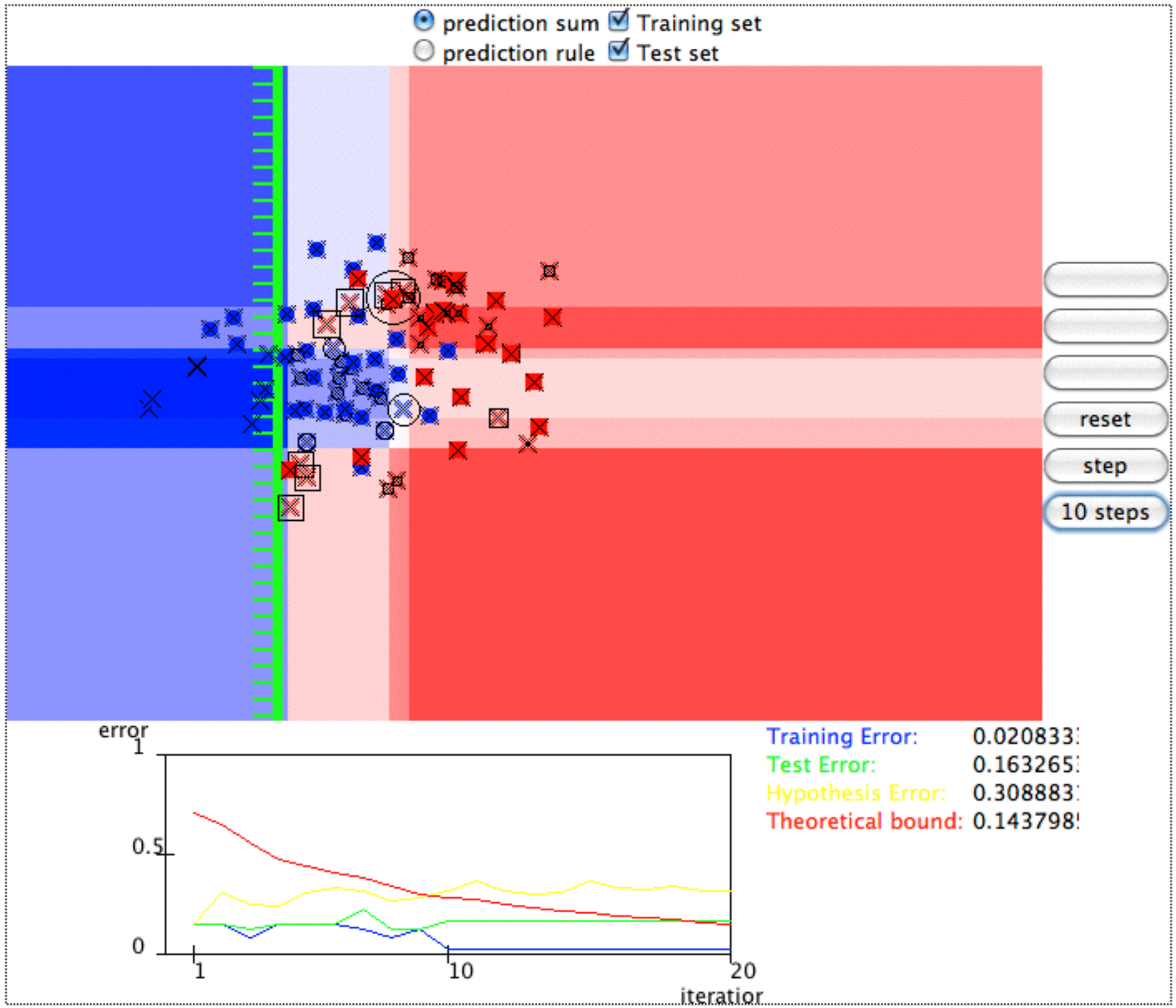
Final Classifier

$$H_{\text{final}} = \text{sign}\left(0.42 \; + 0.65 \; + 0.92 \; \right)$$



$=$



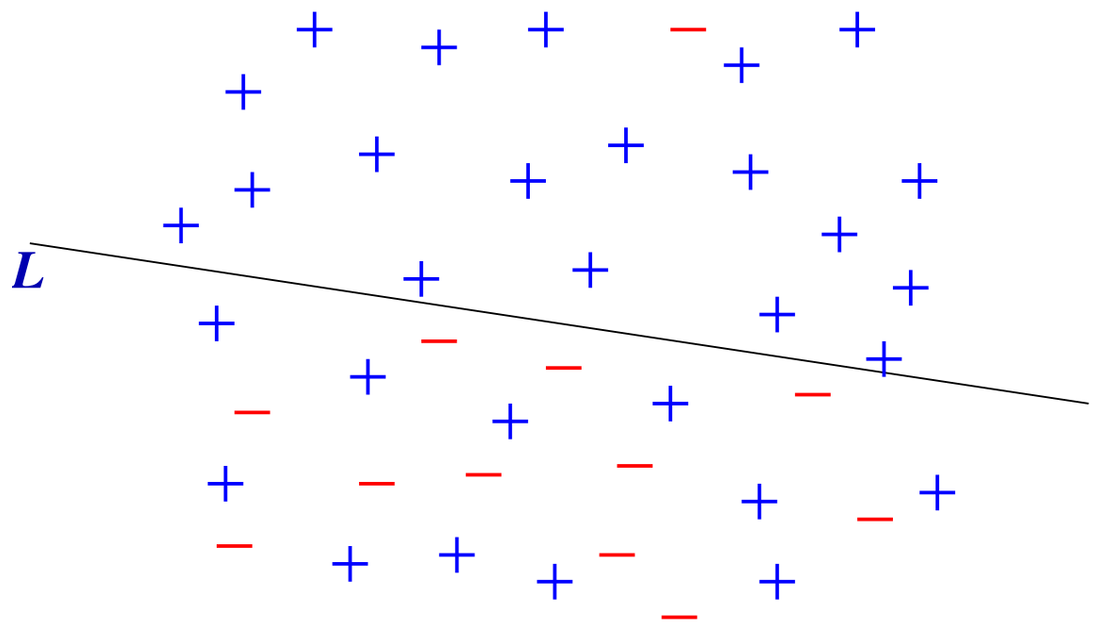


http://cseweb.ucsd.edu/~yfreund/adaboost/index.html

Practical Extensions

- multiclass classification
- ranking problems
- confidence-rated predictions

"Hard" Predictions Can Slow Learning



- ideally, want weak classifier that says:

$$h(x) = \begin{cases} +1 & \text{if } x \text{ above } L \\ \textit{"don't know"} & \text{else} \end{cases}$$

- problem: cannot express using "hard" predictions
- if must predict ± 1 below L, will introduce many "bad" predictions
  - need to "clean up" on later rounds
- dramatically increases time to convergence

Confidence-Rated Predictions

[Schapire & Singer]

- useful to allow weak classifiers to assign confidences to predictions
- formally, allow $h_t : X \longrightarrow \mathbb{R}$

$$\begin{aligned} \text{sign}(h_t(x)) &= \text{prediction} \\ |h_t(x)| &= \text{“confidence”} \end{aligned}$$

- use identical update:

$$D_{t+1}(i) = \frac{D_t(i)}{Z_t} \cdot \exp(-\alpha_t \; y_i \; h_t(x_i))$$

  and identical rule for combining weak classifiers

- question: how to choose α_t and h_t on each round

Confidence-Rated Predictions (cont.)

- saw earlier:

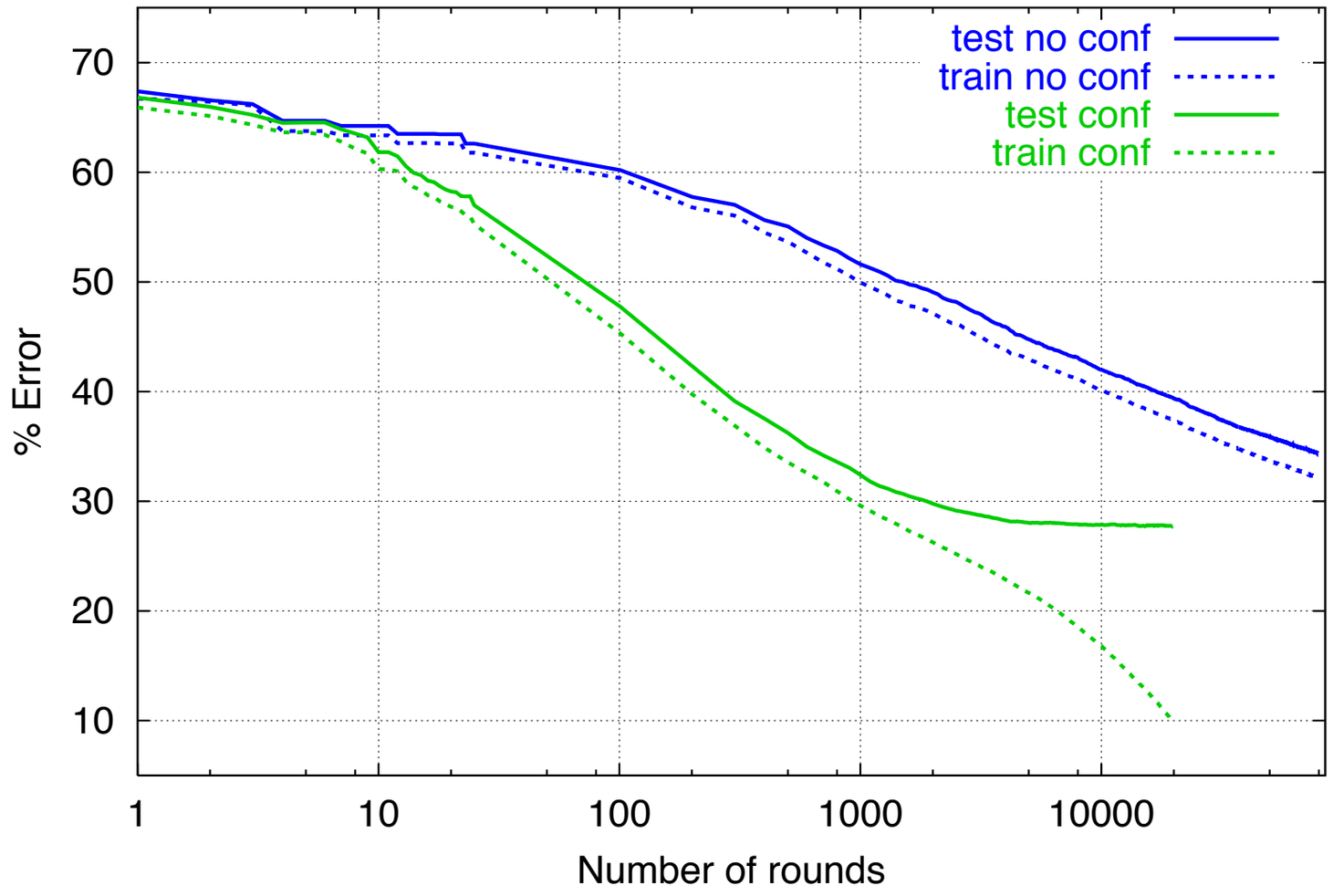
$$\text{training error}(H_{\text{final}}) \leq \prod_t Z_t = \frac{1}{m} \sum_i \exp\left(-y_i \sum_t \alpha_t h_t(x_i)\right)$$

- therefore, on each round t, should choose $\alpha_t h_t$ to minimize:

$$Z_t = \sum_i D_t(i) \exp(-\alpha_t \; y_i \; h_t(x_i))$$

- in many cases (e.g., decision stumps), best confidence-rated weak classifier has simple form that can be found efficiently

Confidence-Rated Predictions Help a Lot



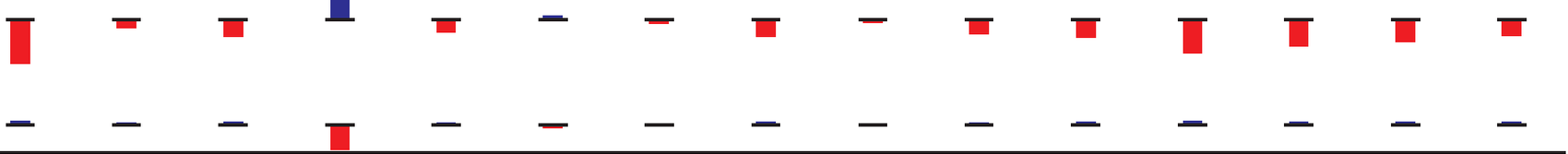
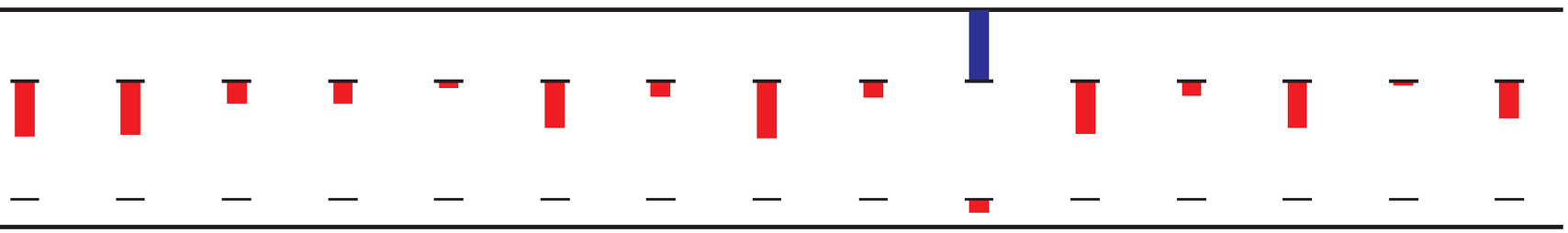
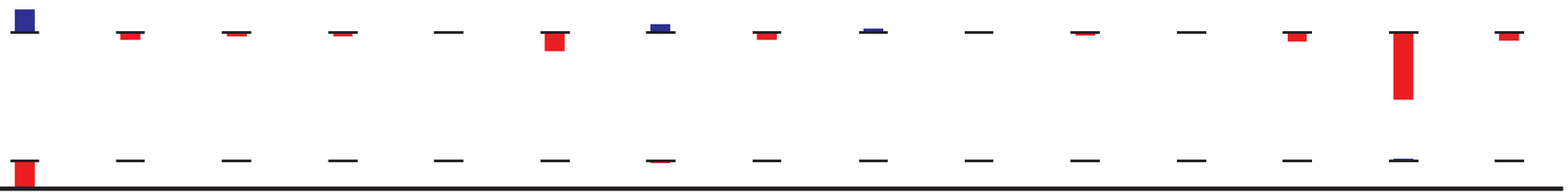
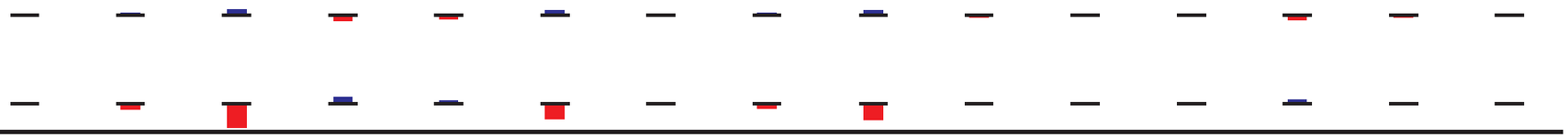
% error	round first reached conf.	no conf.	speedup
40	268	16,938	63.2
35	598	65,292	109.2
30	1,888	>80,000	–

Application: Boosting for Text Categorization

[Schapire & Singer]

- weak classifiers: very simple weak classifiers that test on simple patterns, namely, (sparse) *n*-grams
  - find parameter α_t and rule h_t of given form which minimize Z_t
  - use efficiently implemented exhaustive search
- "How may I help you" data:
  - 7844 training examples
  - 1000 test examples
  - categories: AreaCode, AttService, BillingCredit, CallingCard, Collect, Competitor, DialForMe, Directory, HowToDial, PersonToPerson, Rate, ThirdNumber, Time, TimeCharge, Other.

Weak Classifiers

rnd	term	AC	AS	BC	CC	CO	CM	DM	DI	HO	PP	RA	3N	TI	TC	OT
1	collect															
2	card															
3	my home															
4	person ? person															
5	code															
6	I															

More Weak Classifiers

rnd	term	AC	AS	BC	CC	CO	CM	DM	DI	HO	PP	RA	3N	TI	TC	OT
7	time															
8	wrong number															
9	how															
10	call															
11	seven															
12	trying to															
13	and															

More Weak Classifiers

rnd	term	AC	AS	BC	CC	CO	CM	DM	DI	HO	PP	RA	3N	TI	TC	OT
14	third															
15	to															
16	for															
17	charges															
18	dial															
19	just															

Finding Outliers

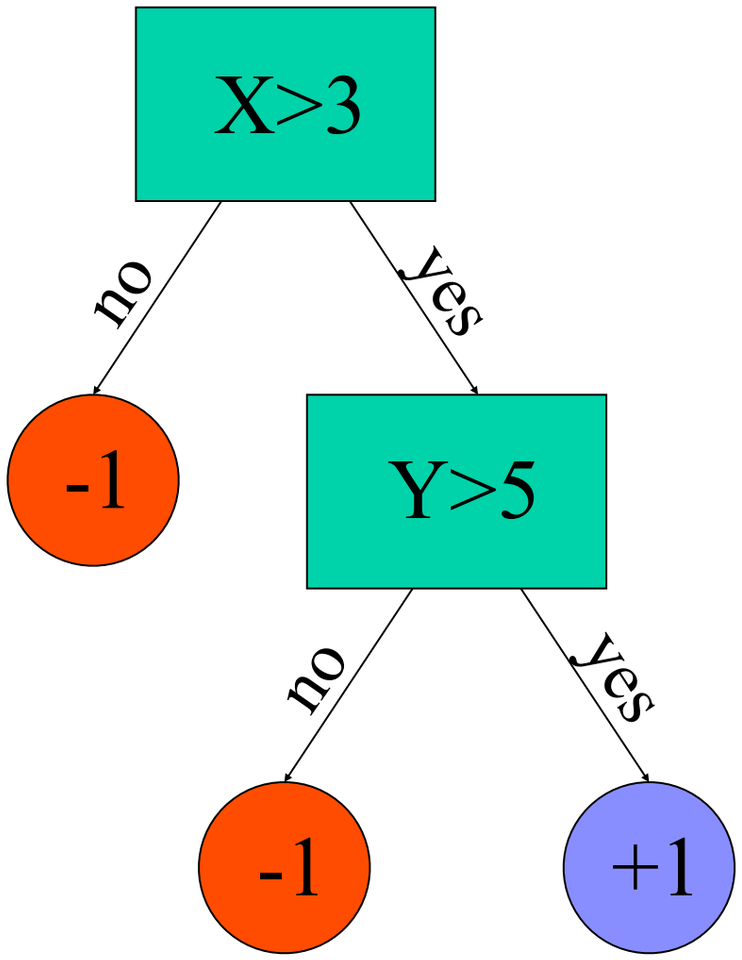
examples with most weight are often outliers (mislabeled and/or ambiguous)

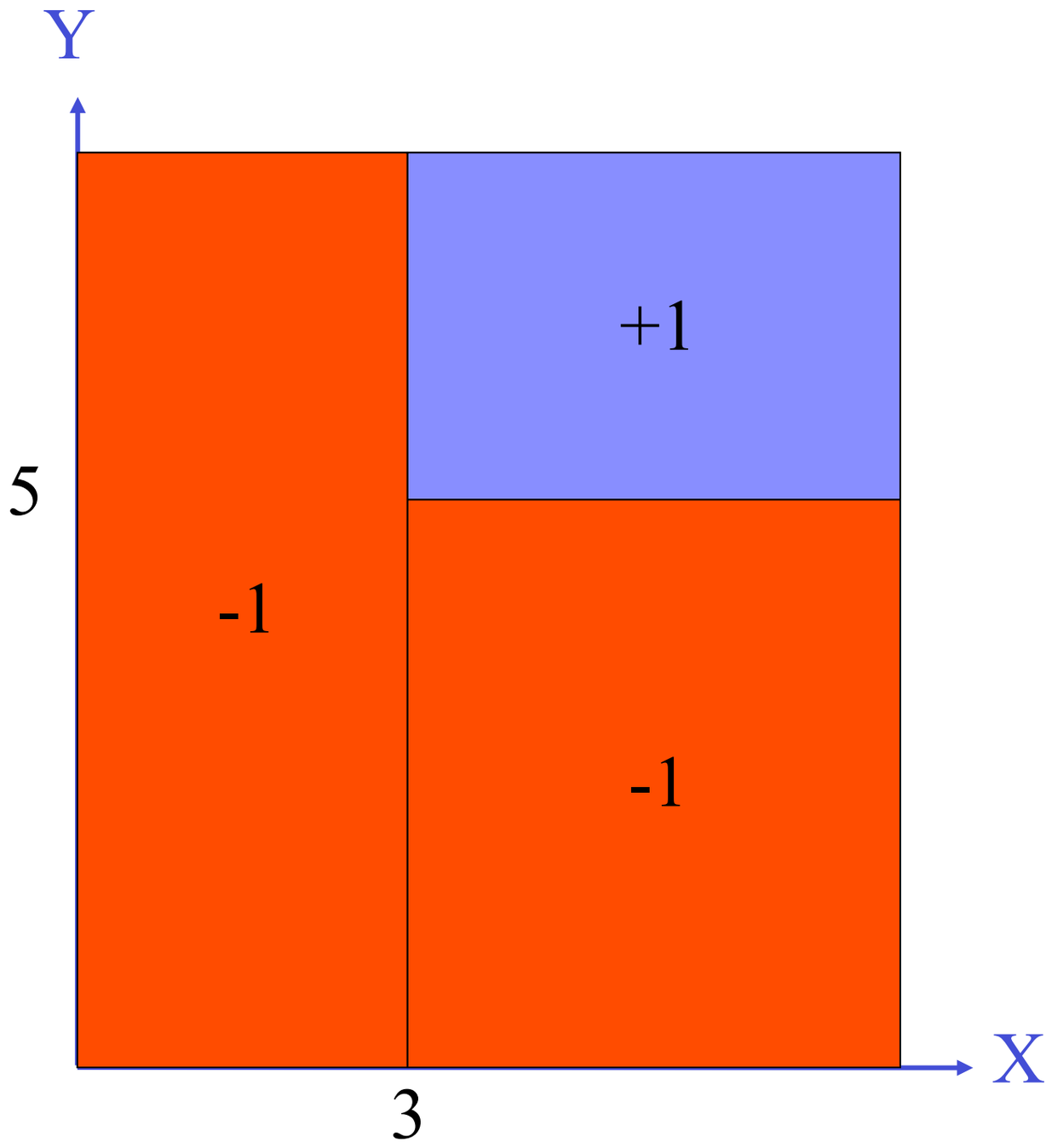
- `I'm trying to make a credit card call` (Collect)
- `hello` (Rate)
- `yes I'd like to make a long distance collect call please` (CallingCard)
- `calling card please` (Collect)
- `yeah I'd like to use my calling card number` (Collect)
- `can I get a collect call` (CallingCard)
- `yes I would like to make a long distant telephone call and have the charges billed to another number` (CallingCard DialForMe)
- `yeah I can not stand it this morning I did oversea call is so bad` (BillingCredit)
- `yeah special offers going on for long distance` (AttService Rate)
- `mister allen please william allen` (PersonToPerson)
- `yes ma'am I I'm trying to make a long distance call to a non dialable point in san miguel philippines` (AttService Other)

Alternating Decision Trees

With Llew Mason

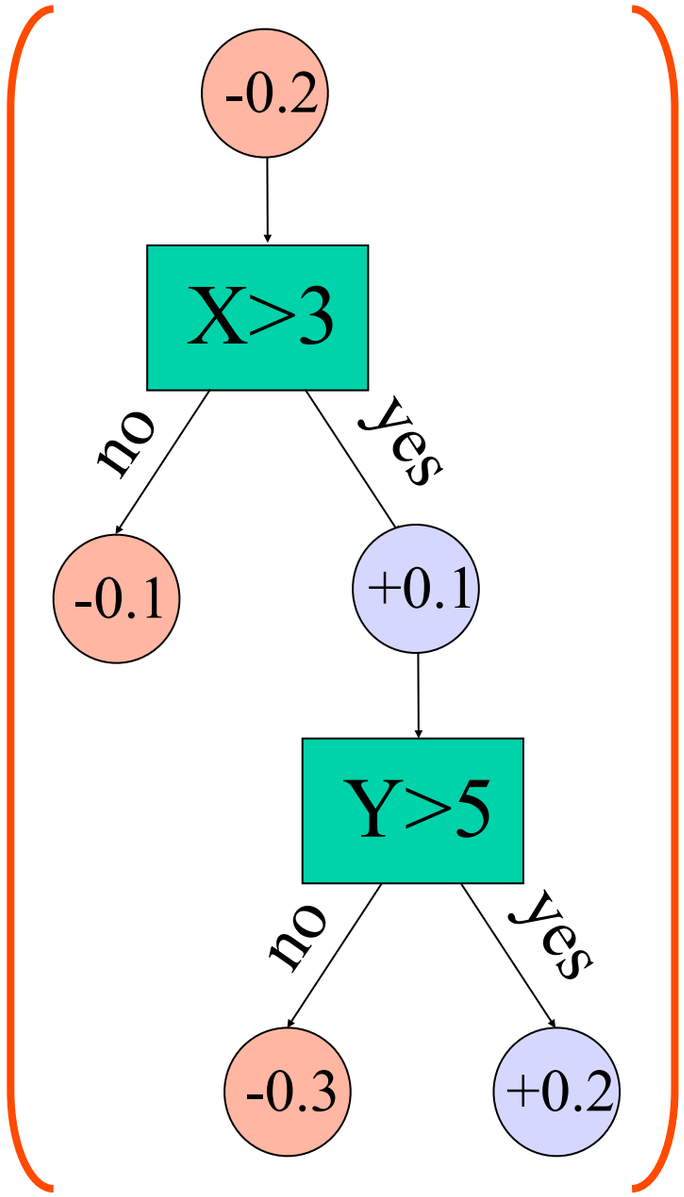
Decision Trees

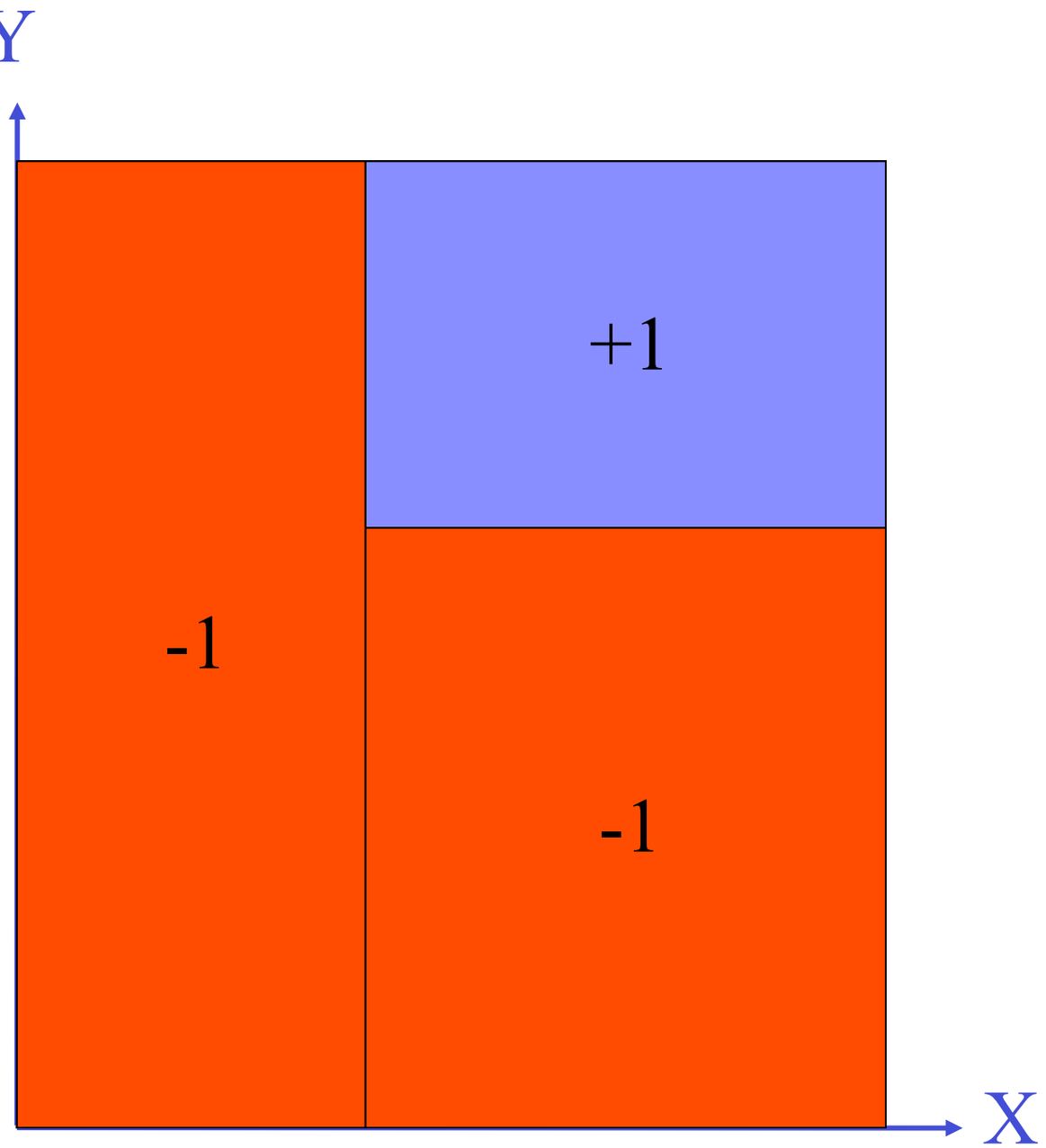




Decision tree as a sum

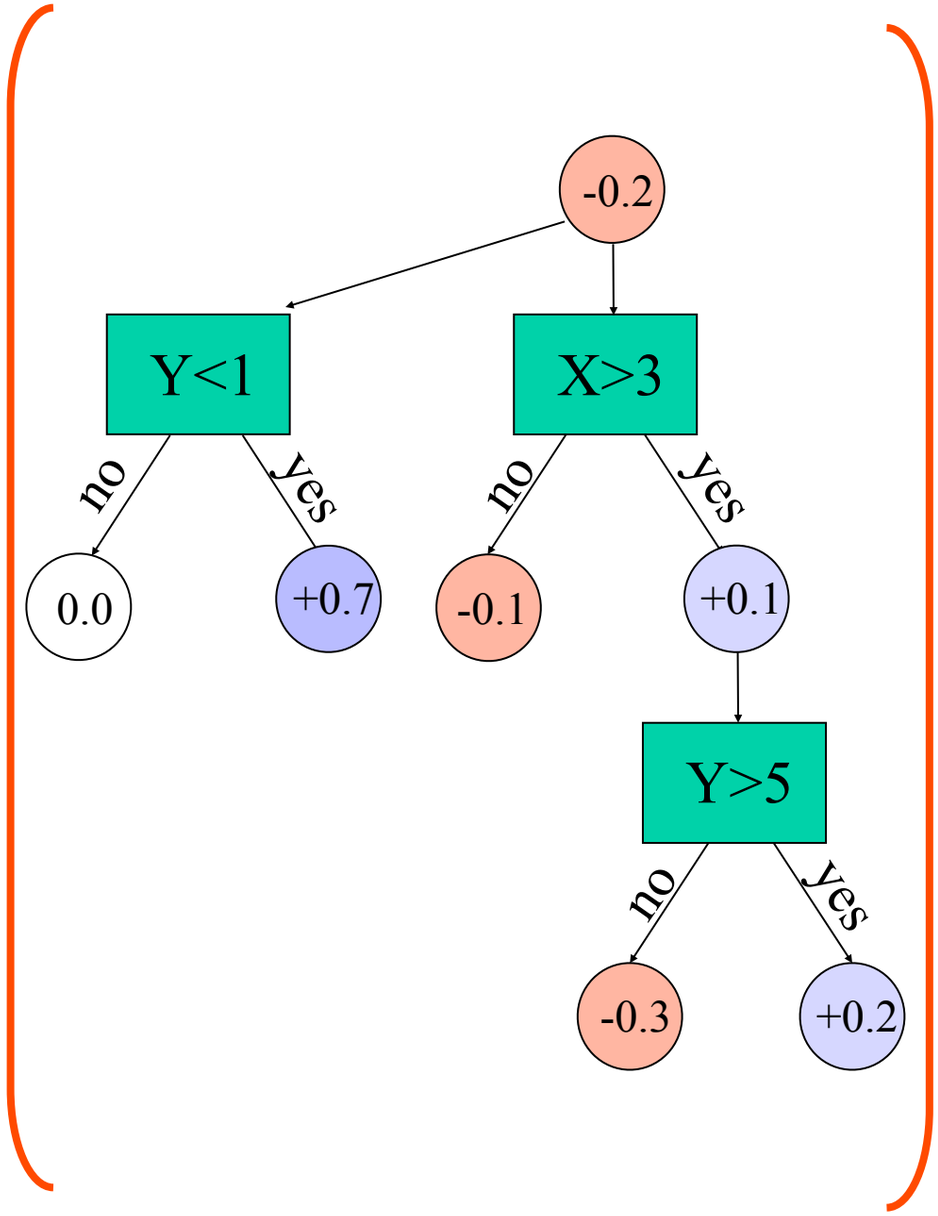
sign

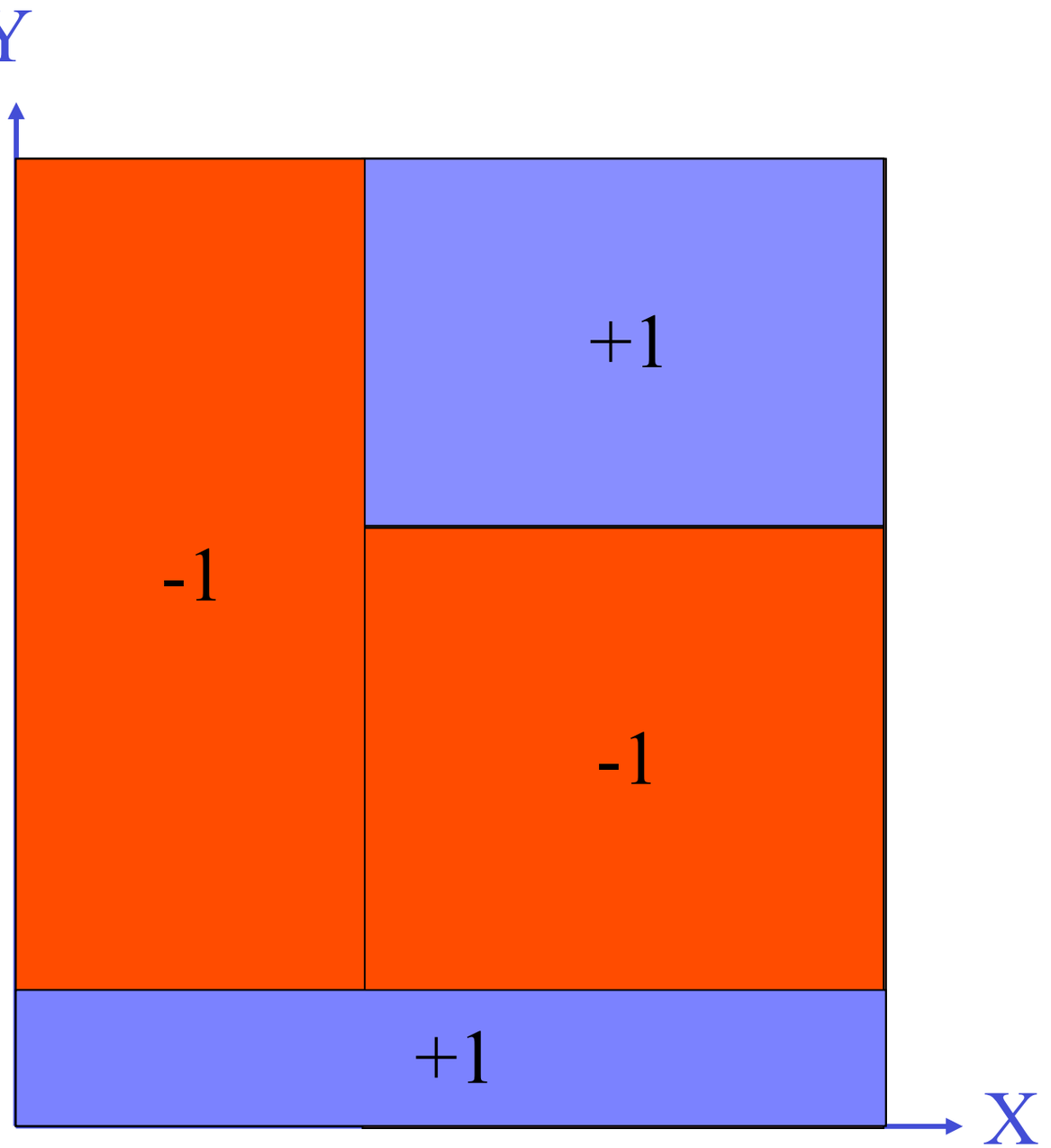




An alternating decision tree

sign





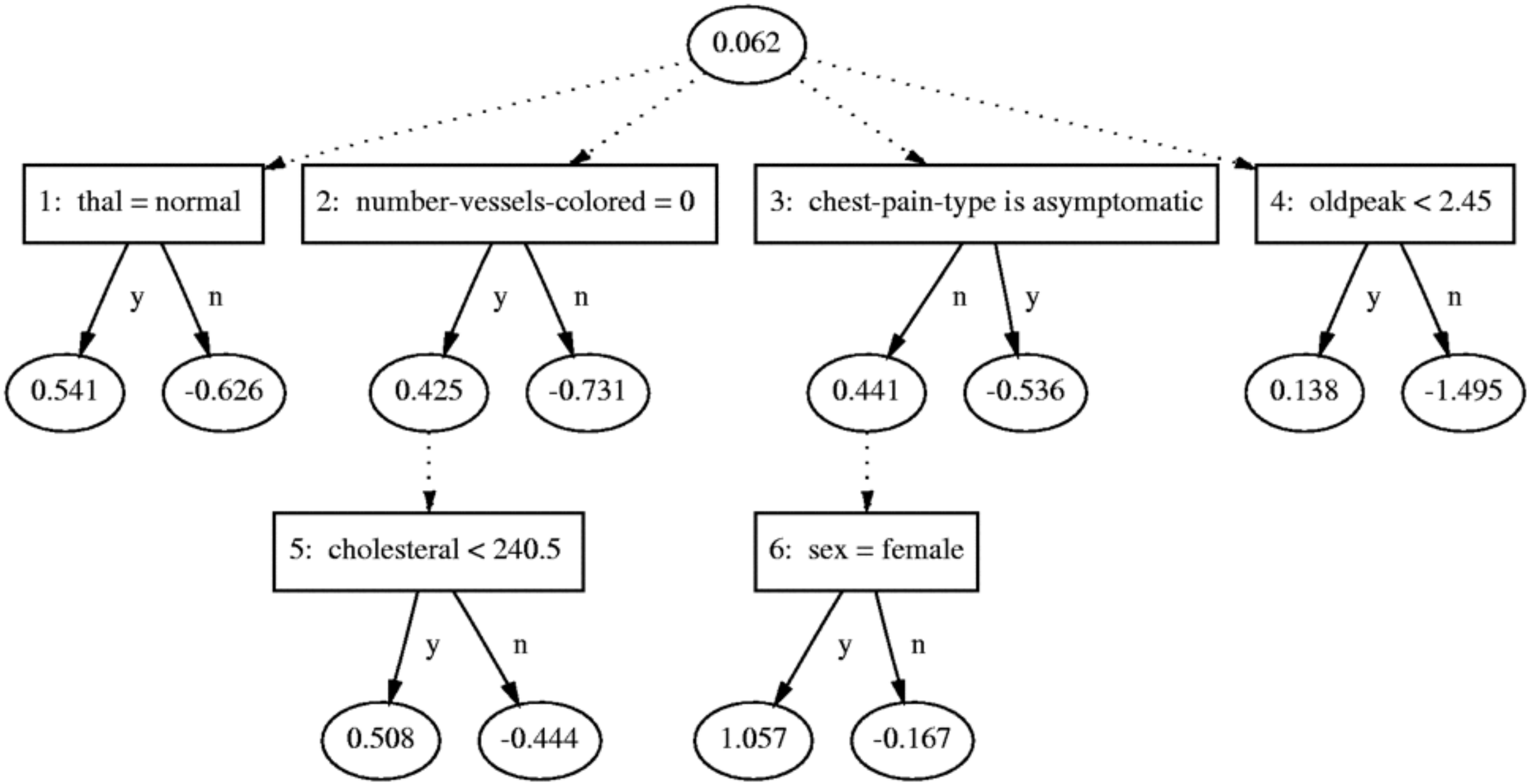
Example: Medical Diagnostics

- `Cleve` dataset from UC Irvine database.
- Heart disease diagnostics (+1=healthy,-1=sick)
- 13 features from tests (real valued and discrete).
- 303 instances.

Cross-validated accuracy

Learning algorithm	Number of splits	Average test error	Test error variance
ADtree	6	17.0%	0.6%
C5.0	27	27.2%	0.5%
C5.0 + boosting	446	20.2%	0.5%
Boost Stumps	16	16.5%	0.8%

Adtree for Cleveland heart-disease diagnostics problem



Freund, Mason, Rogers, Pregibon, Cortes 2000

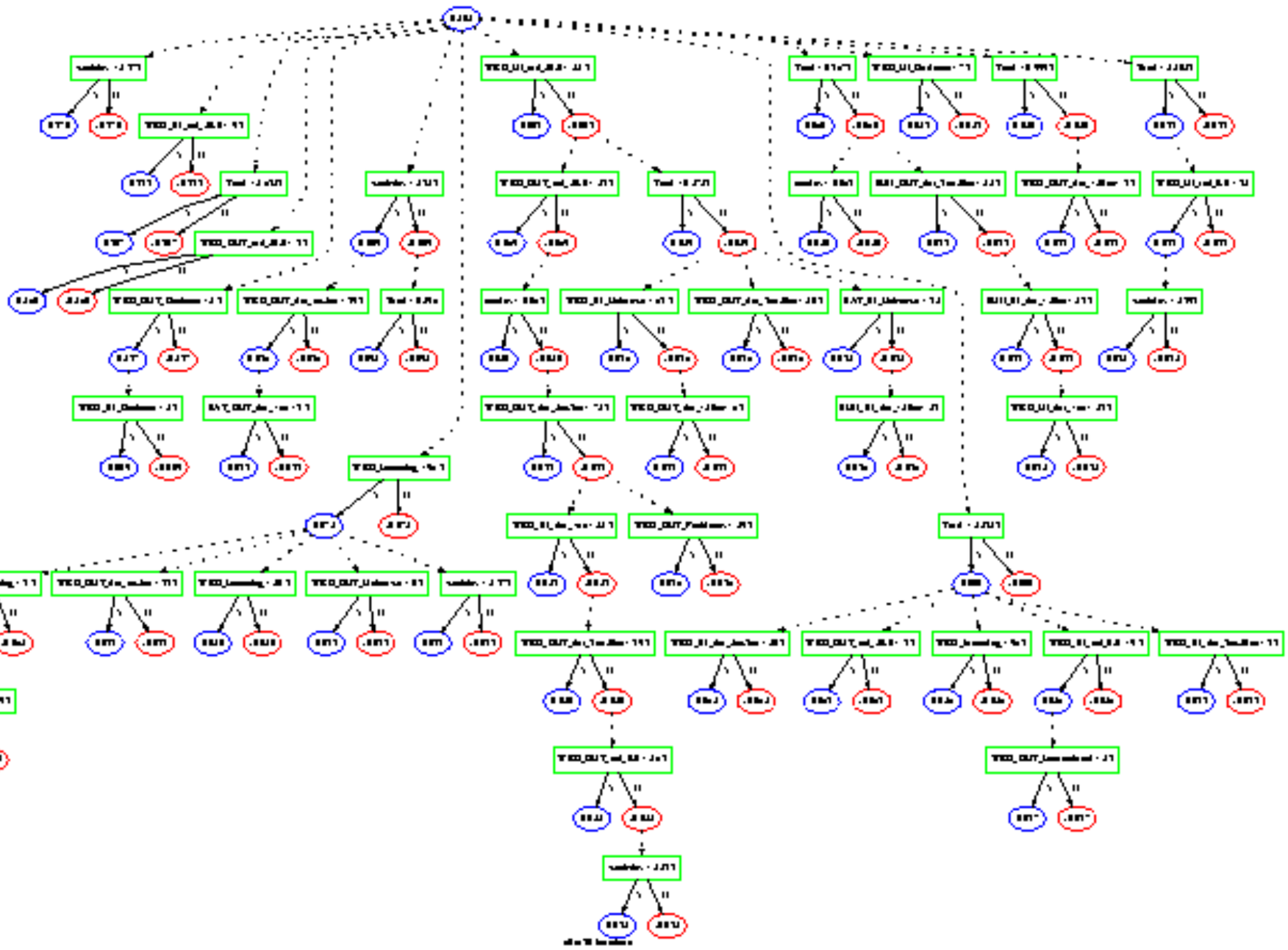
Call Detail analysis (AT&T)

- Distinguish business/residence customers
- Using statistics from call-detail records
  - Label unknown for ~30% of phone numbers
  - Reason: mostly because local phone companies collect information but sometimes don't share it with AT&T...

Massive datasets

- 260M calls / day
- 230M telephone numbers
- Hancock: software for computing statistical signatures
  (today we might have used Hadoop)
- 100K randomly selected training examples,
- ~10K is enough
  - Training takes about 2 hours.
  - Generated classifier has to be both accurate and efficient

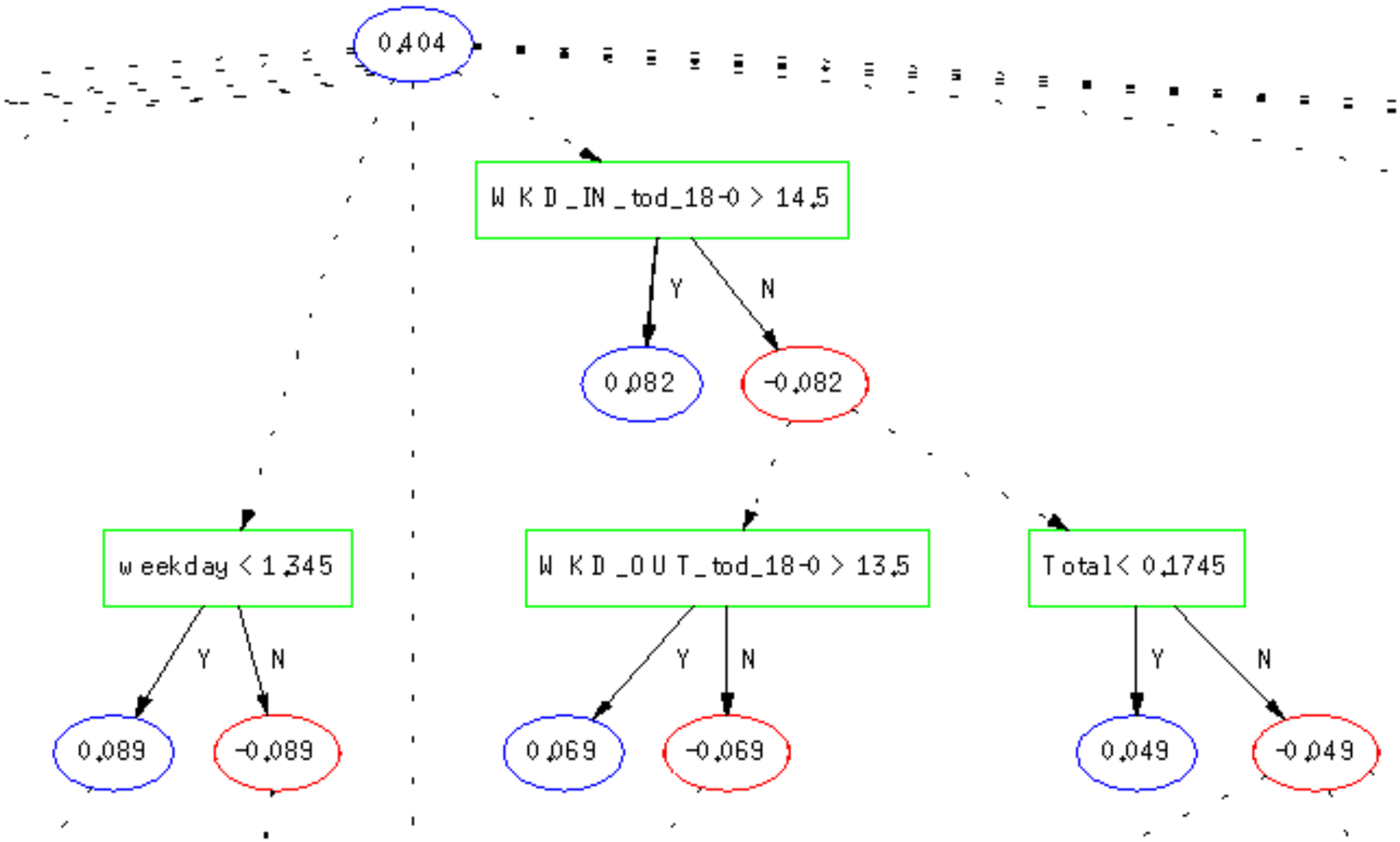
Alternating tree for “buizocity”



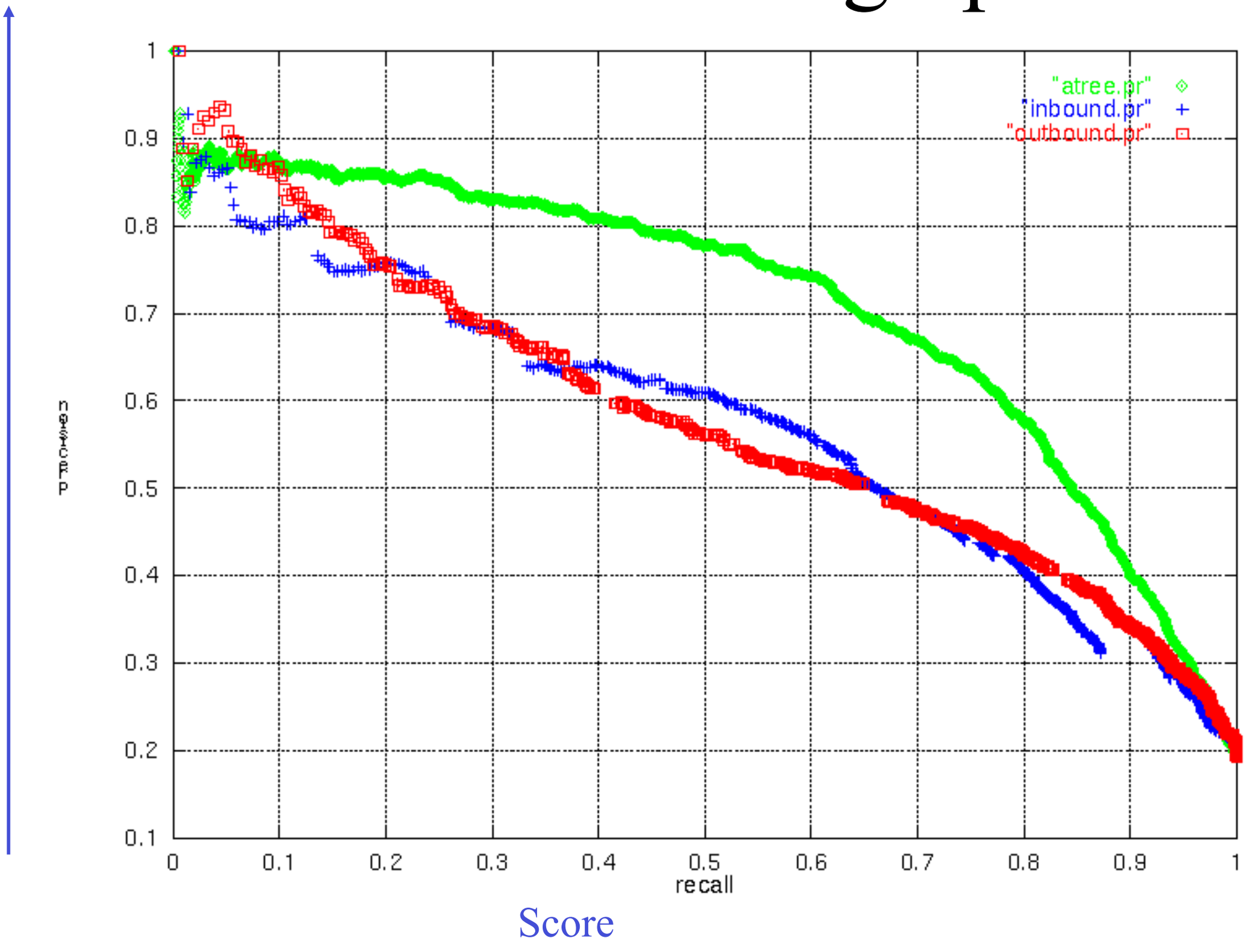
Alternating Tree (Detail)

Positive predictions ⇔ Residences

Negative predictions ⇔ Businesses

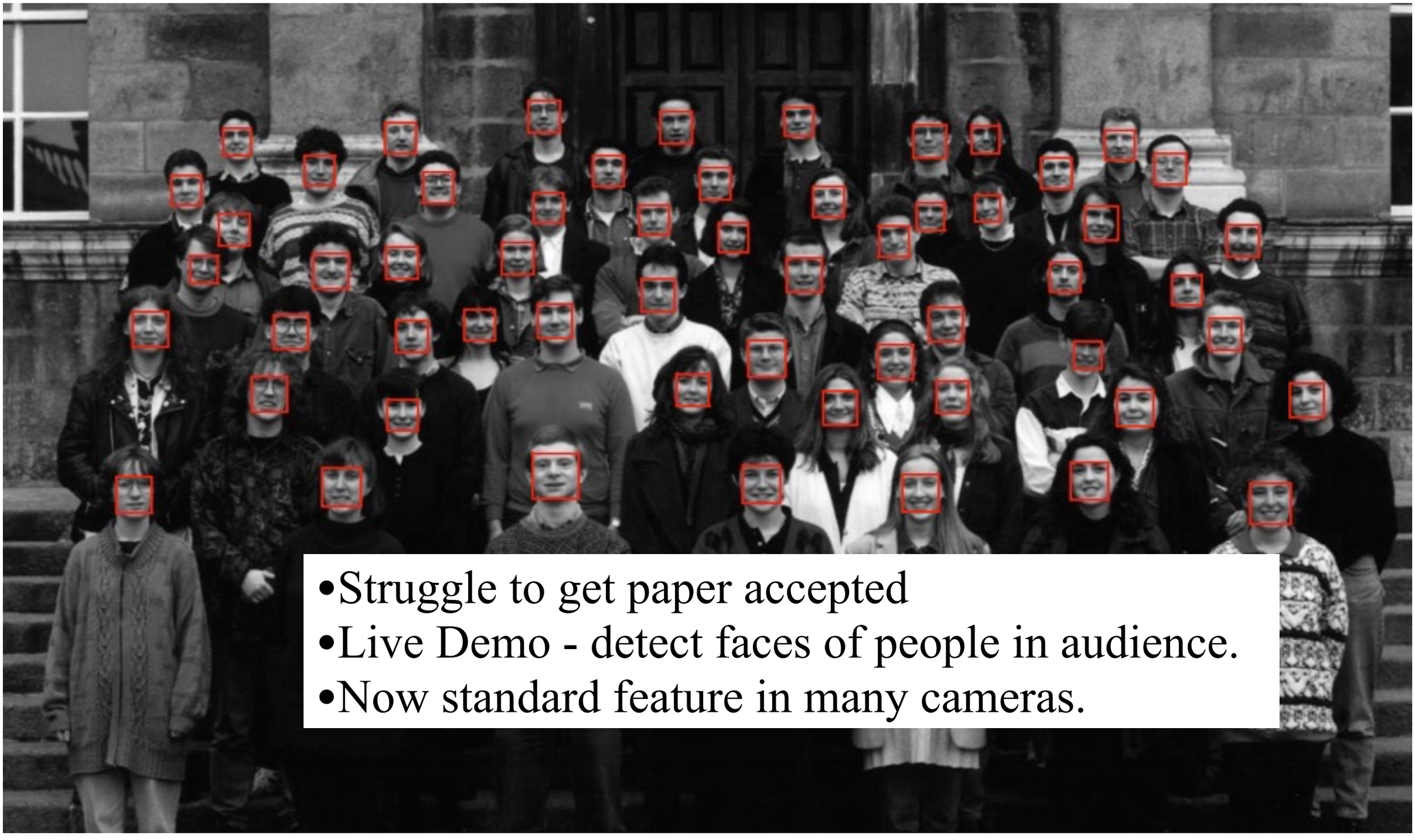


Precision/recall graphs



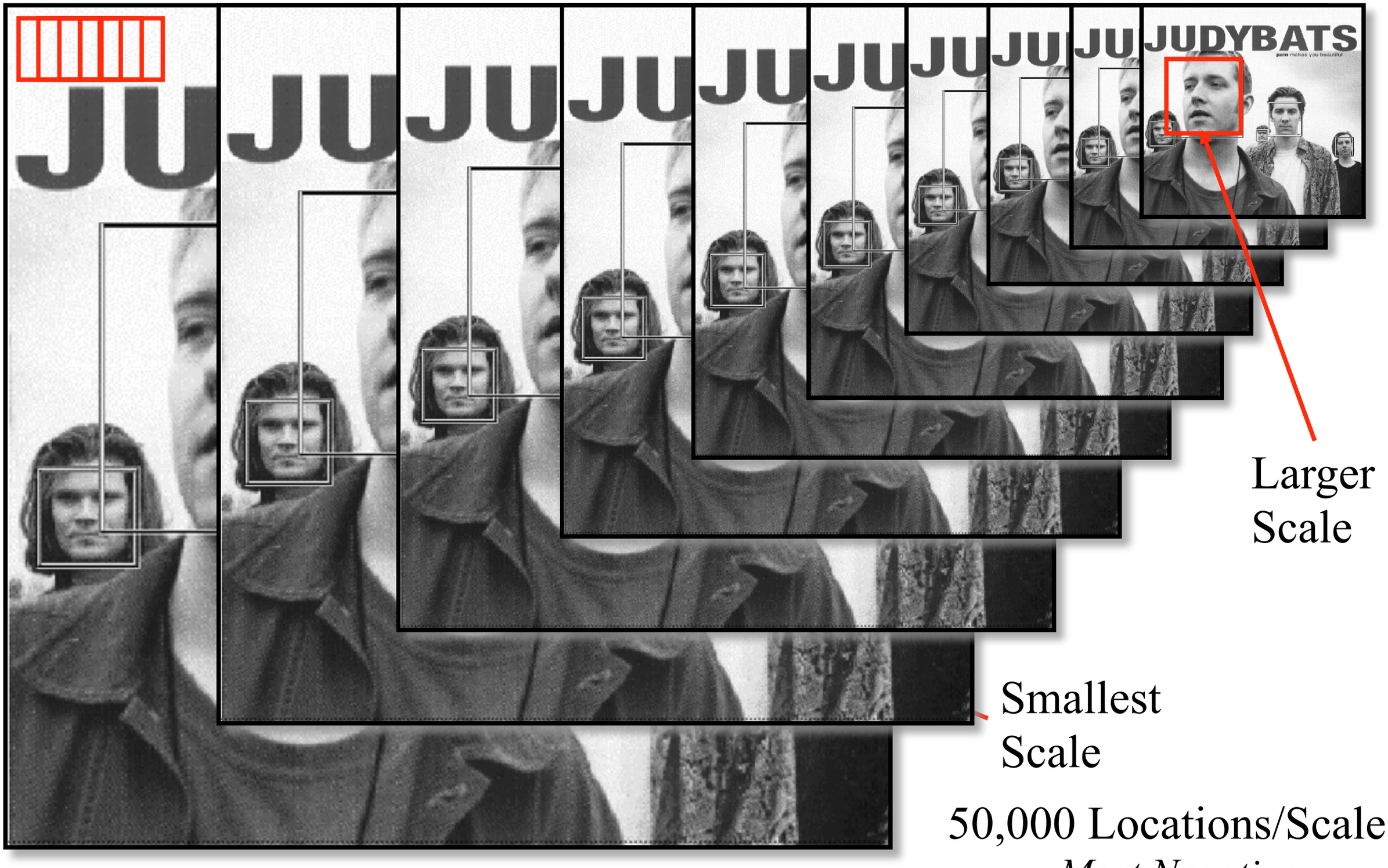
Viola and Jones face detector

Face Detection / Viola and Jones

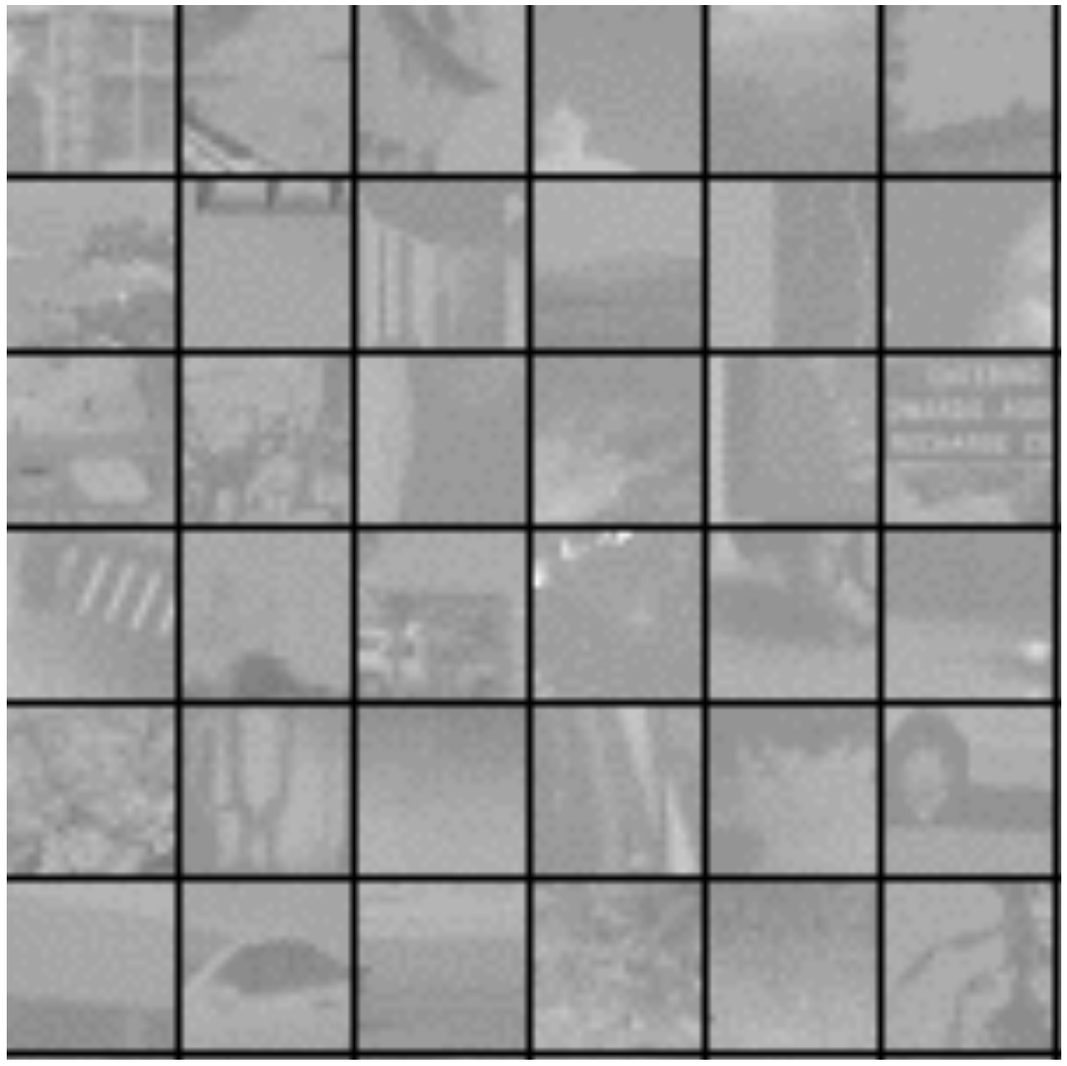


- Struggle to get paper accepted
- Live Demo - detect faces of people in audience.
- Now standard feature in many cameras.

Face Detection as a Filtering process



Classifier is Learned from Labeled Data



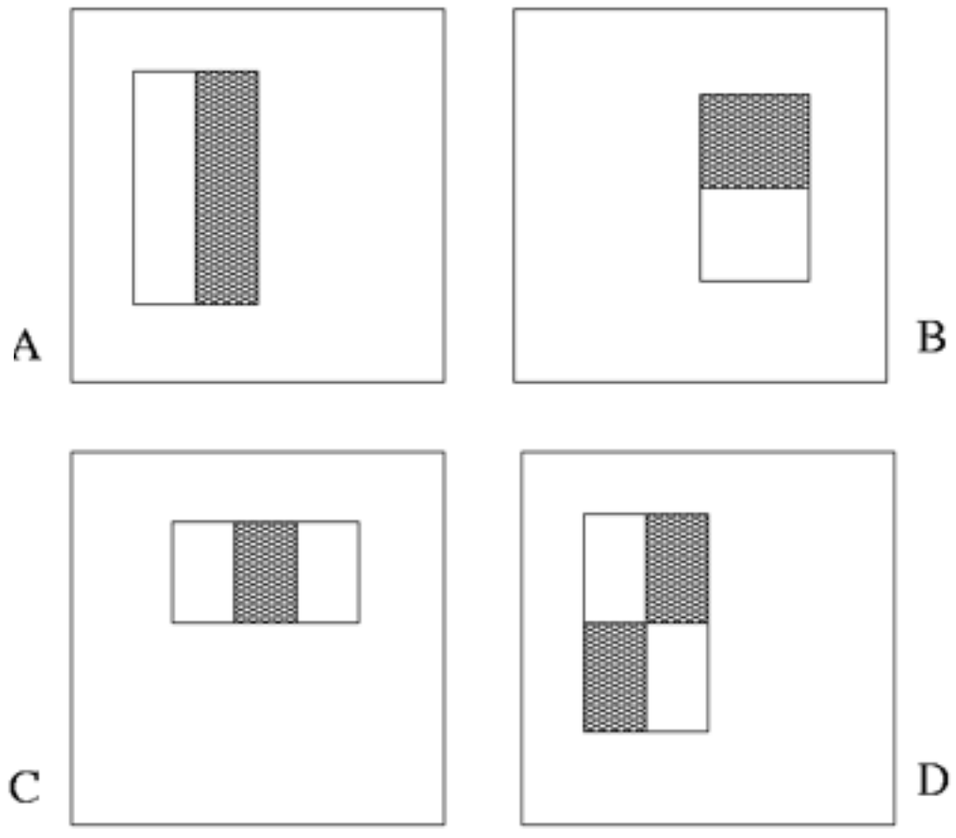
- 5000 faces, 10^8 non faces
- Faces are normalized
  - Scale, translation
  - Rotation remains…

Image Features

"Rectangle filters"

Similar to Haar wavelets

Papageorgiou, et al.



$$h_t(x_i) = \begin{cases} 1 & \text{if } f_t(x_i) > \theta_t \\ 0 & \text{otherwise} \end{cases}$$

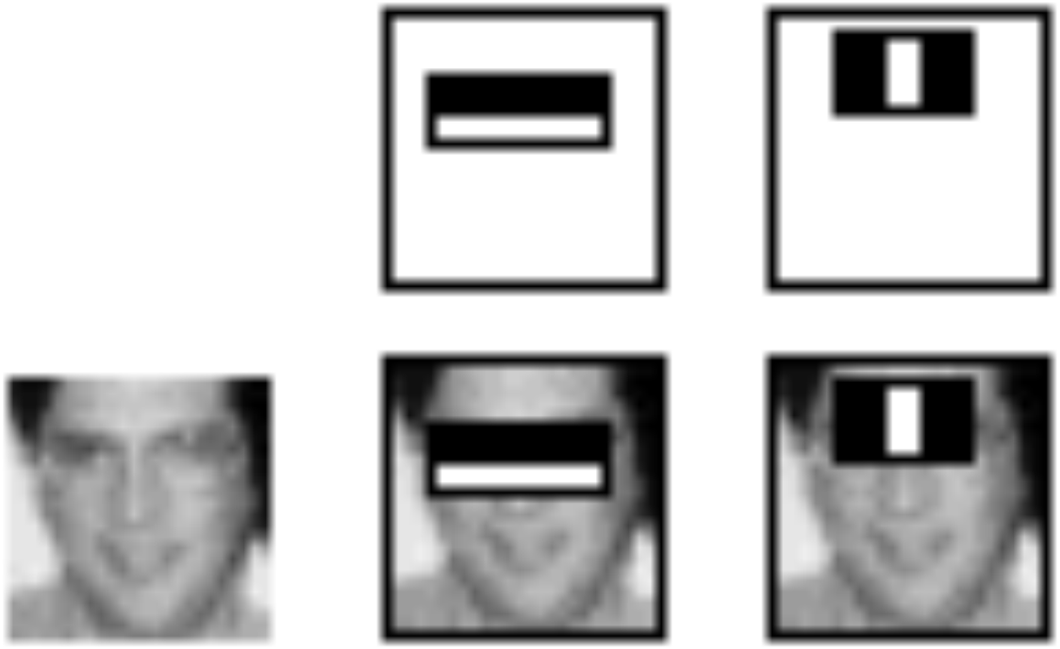
Very fast to compute using "integral image".

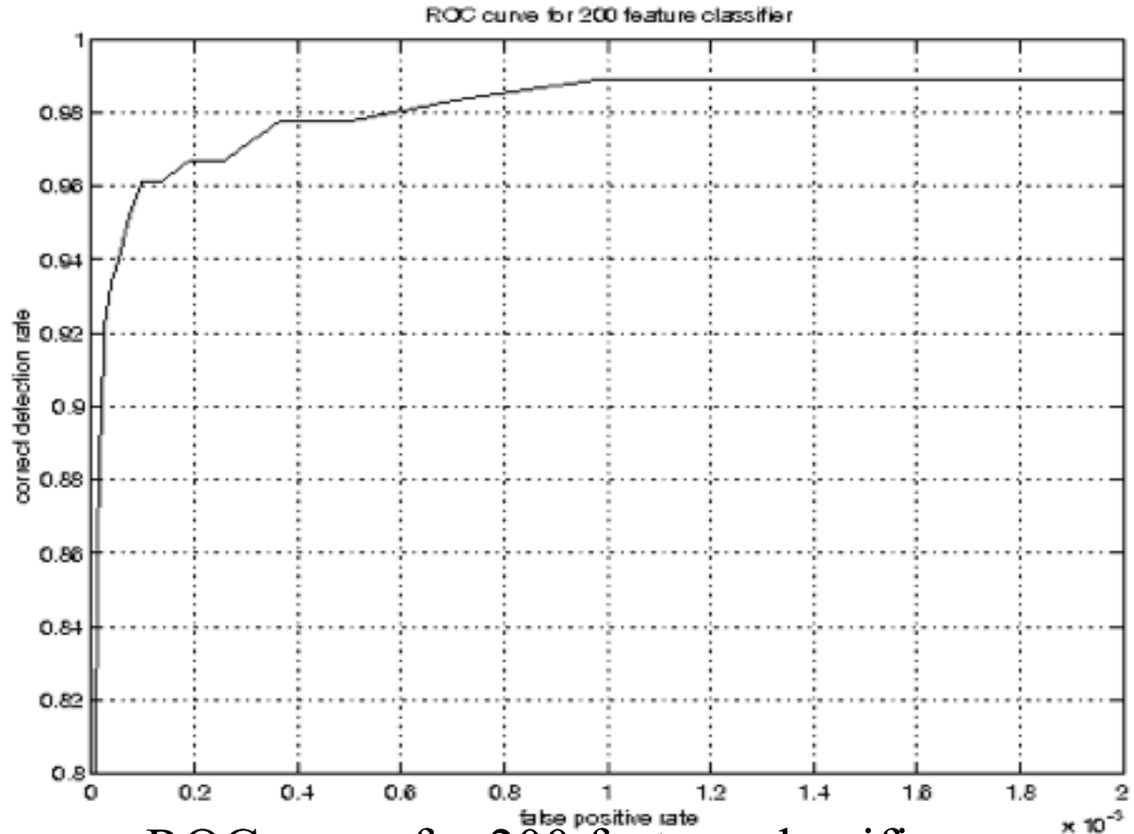
$60{,}000 \times 100 = 6{,}000{,}000$
Unique Features

Combined using adaboost

Example Classifier for Face Detection

- A classifier with 200 rectangle features was learned using AdaBoost
- 95% correct detection on test set with 1 in 14084 false positives.
- To be competitive, needs ~6,000 features
- But that makes detector prohibitively slow.
- Learning is always slow, but done only once..



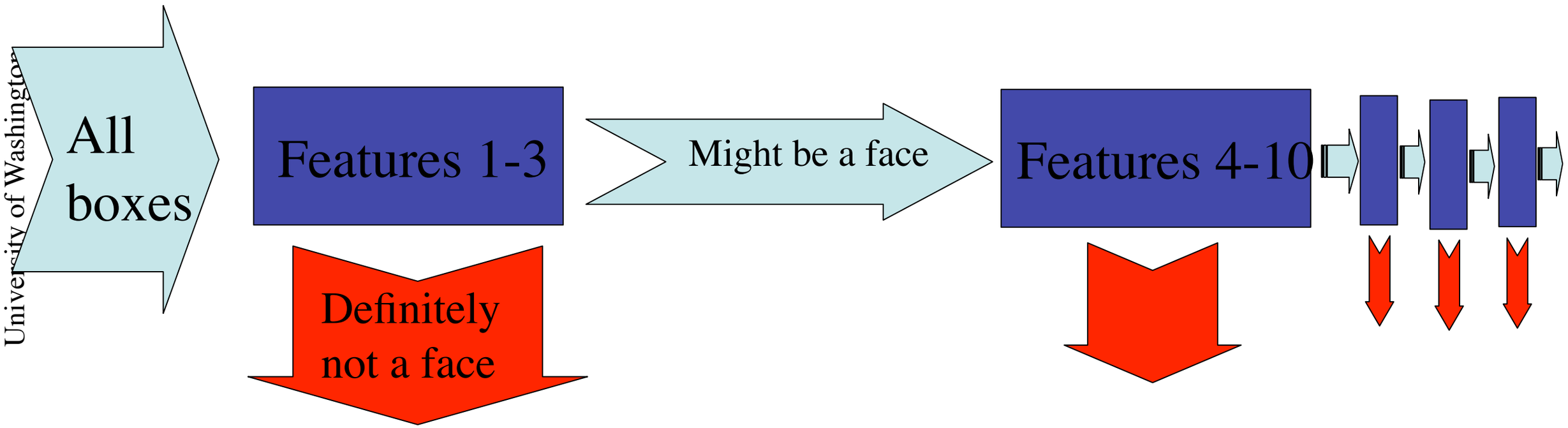


ROC curve for 200 feature classifier

Employing a cascade to minimize average feature computation time

The accurate detector combines 6000 simple features using Adaboost.

In most boxes, only 8-9 features are calculated.



Co-Training

Opera Solutions, 1/20/2012

Using confidence to avoid labeling

Levin, Viola, Freund 2003

Image 1

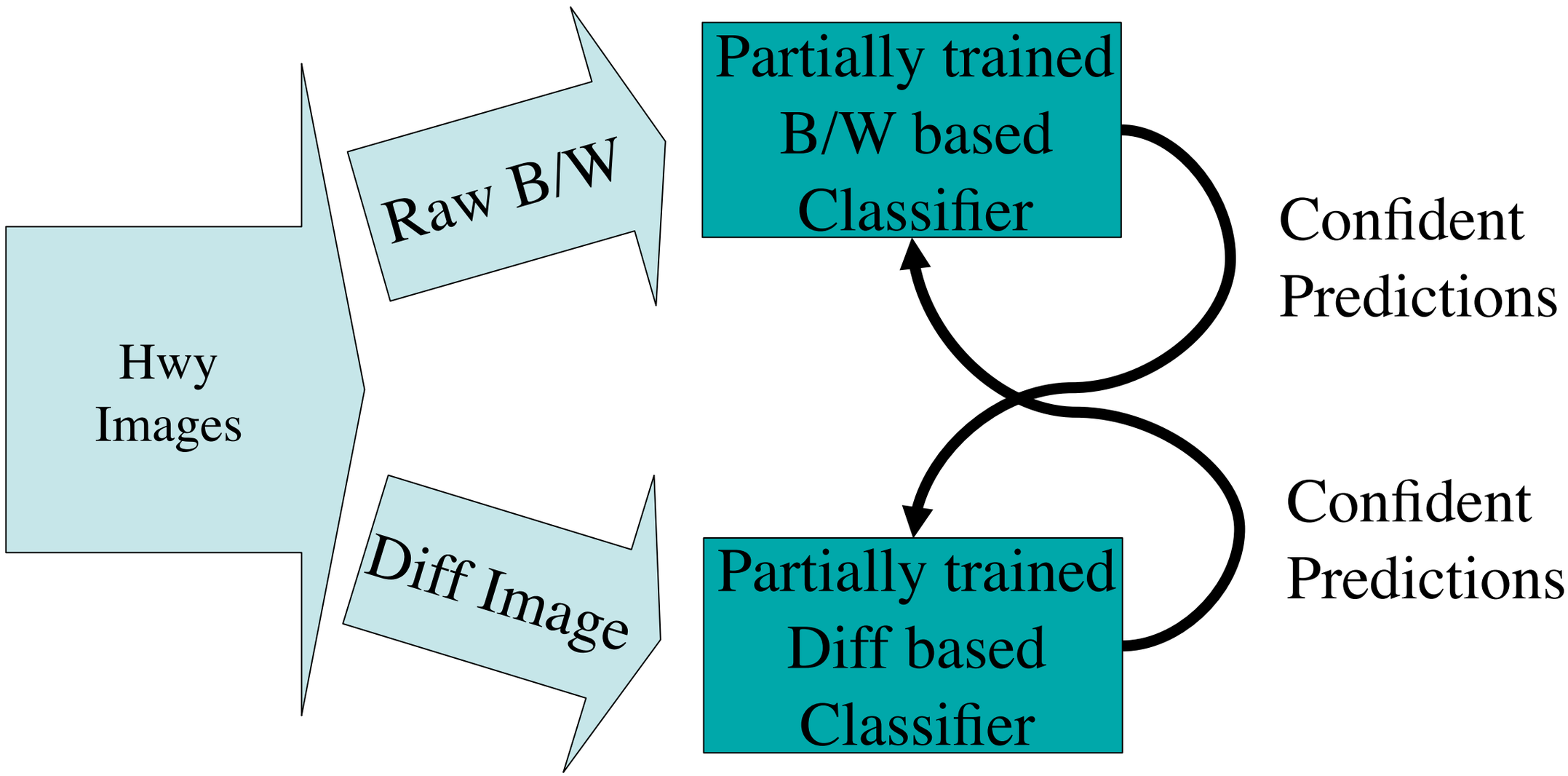
Image 1 - diff from time average

Image 2

Image 2 - diff from time average

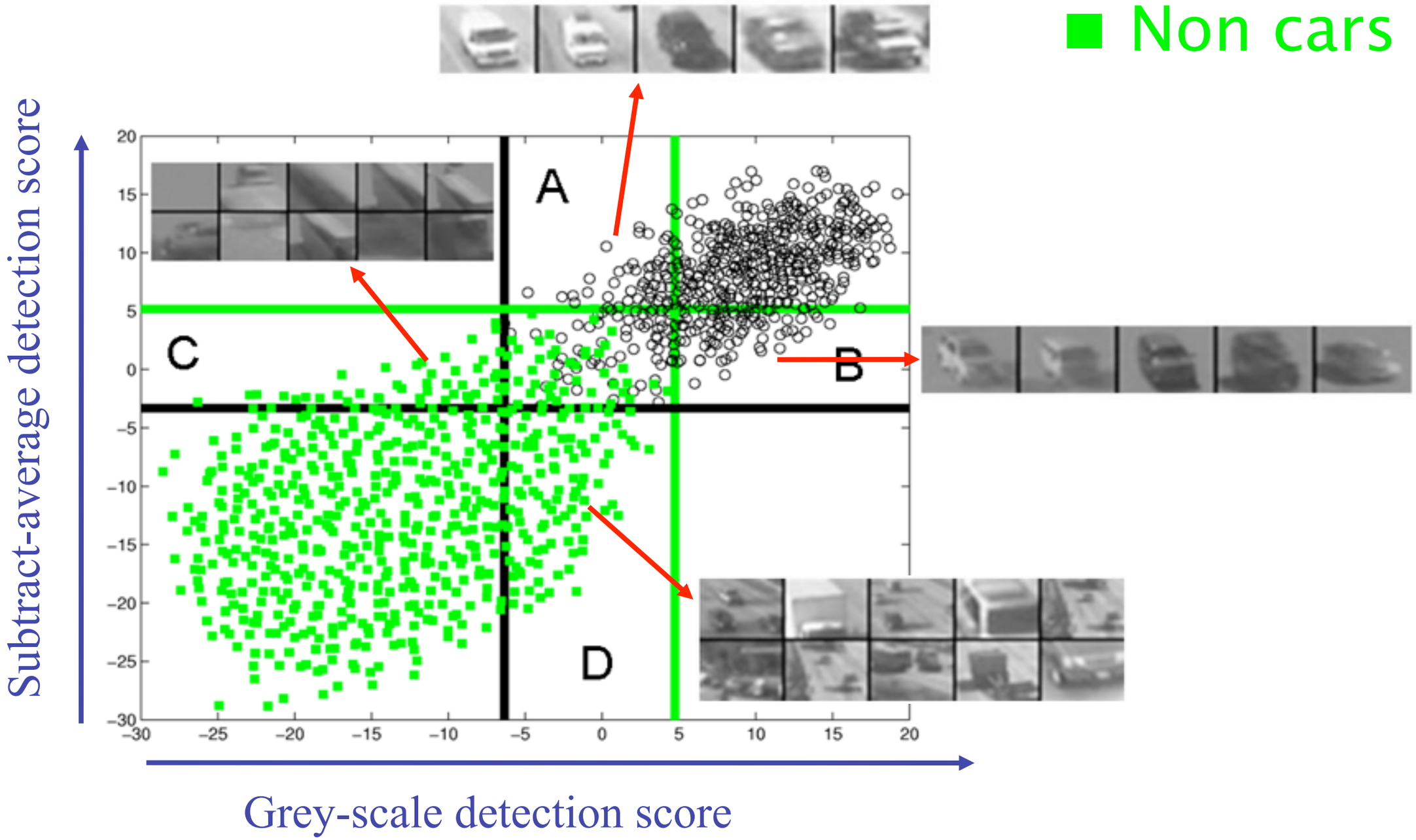
Co-training

Blum and Mitchell 98



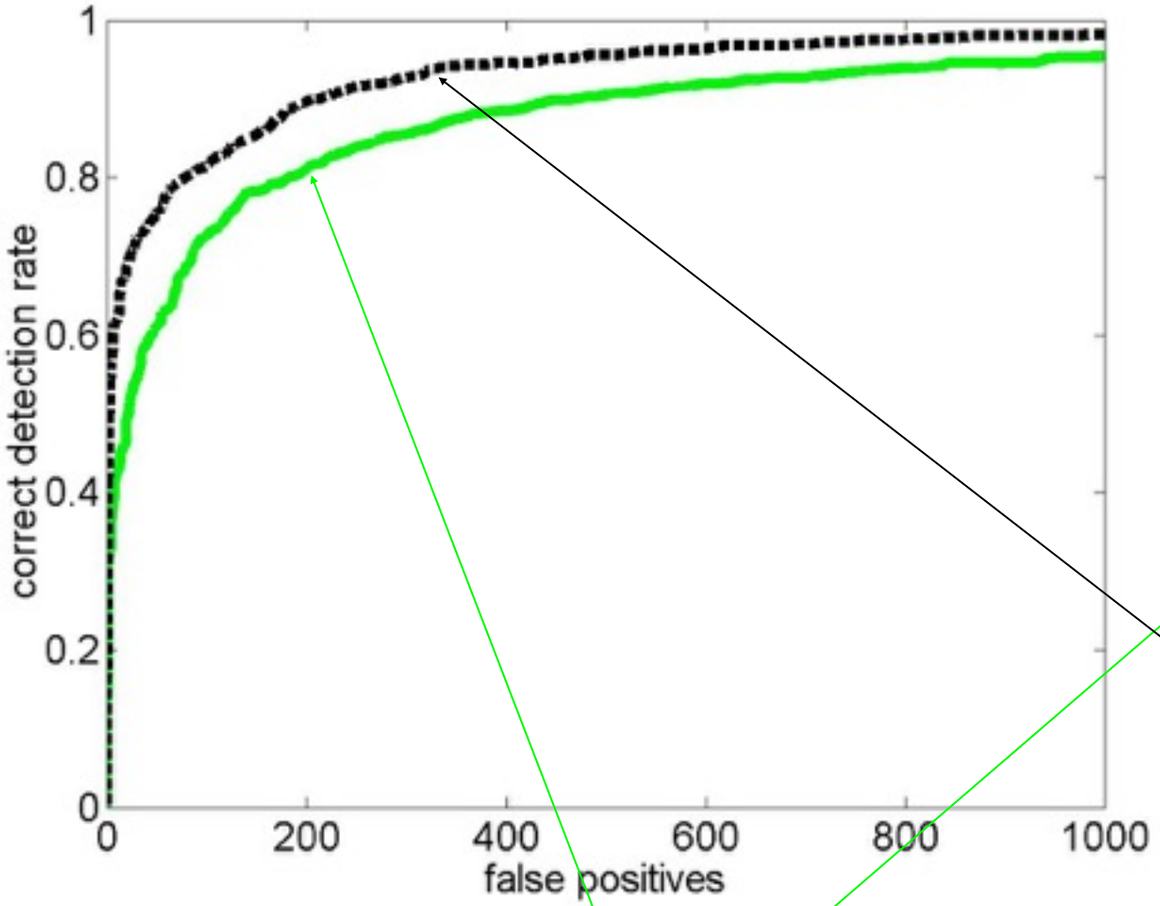
○ Cars

■ Non cars

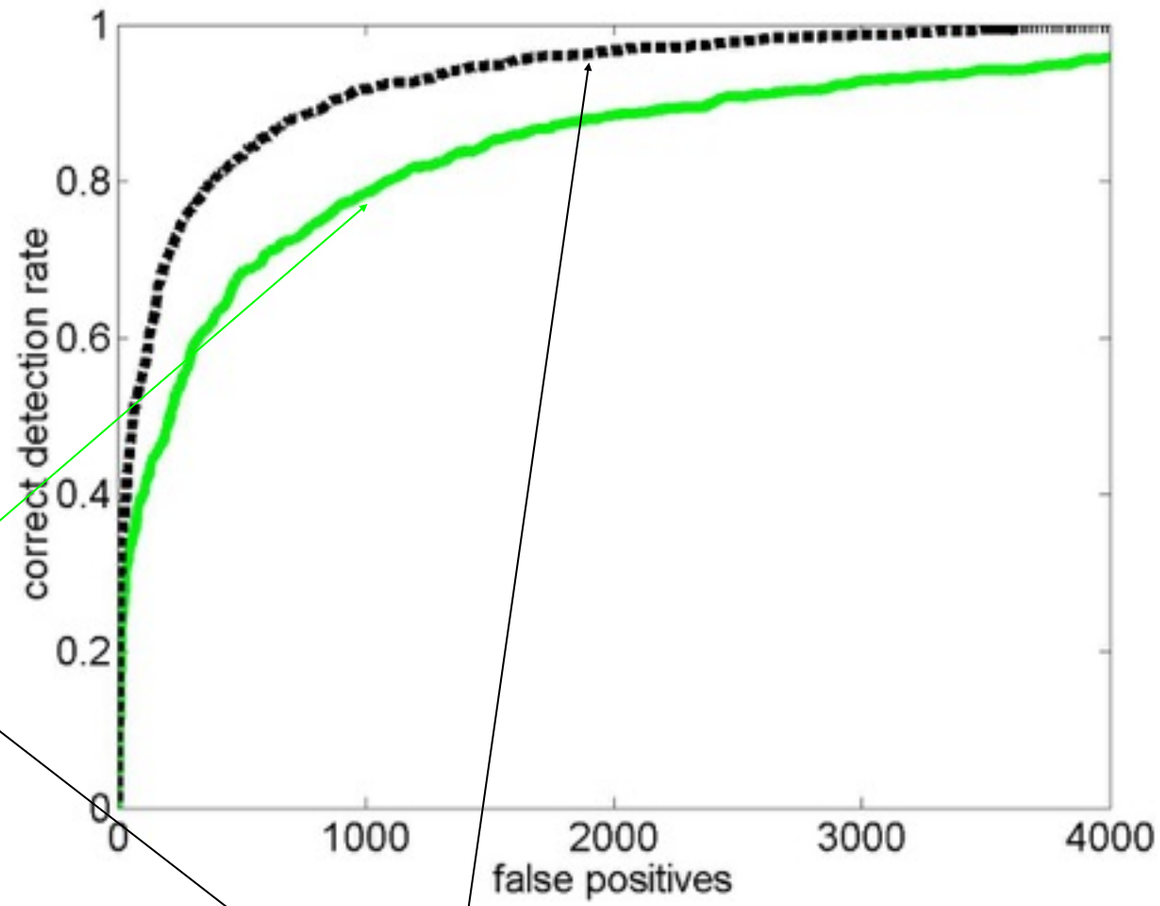


Co-Training Results

Raw Image detector



Difference Image detector



Un

Before co-training

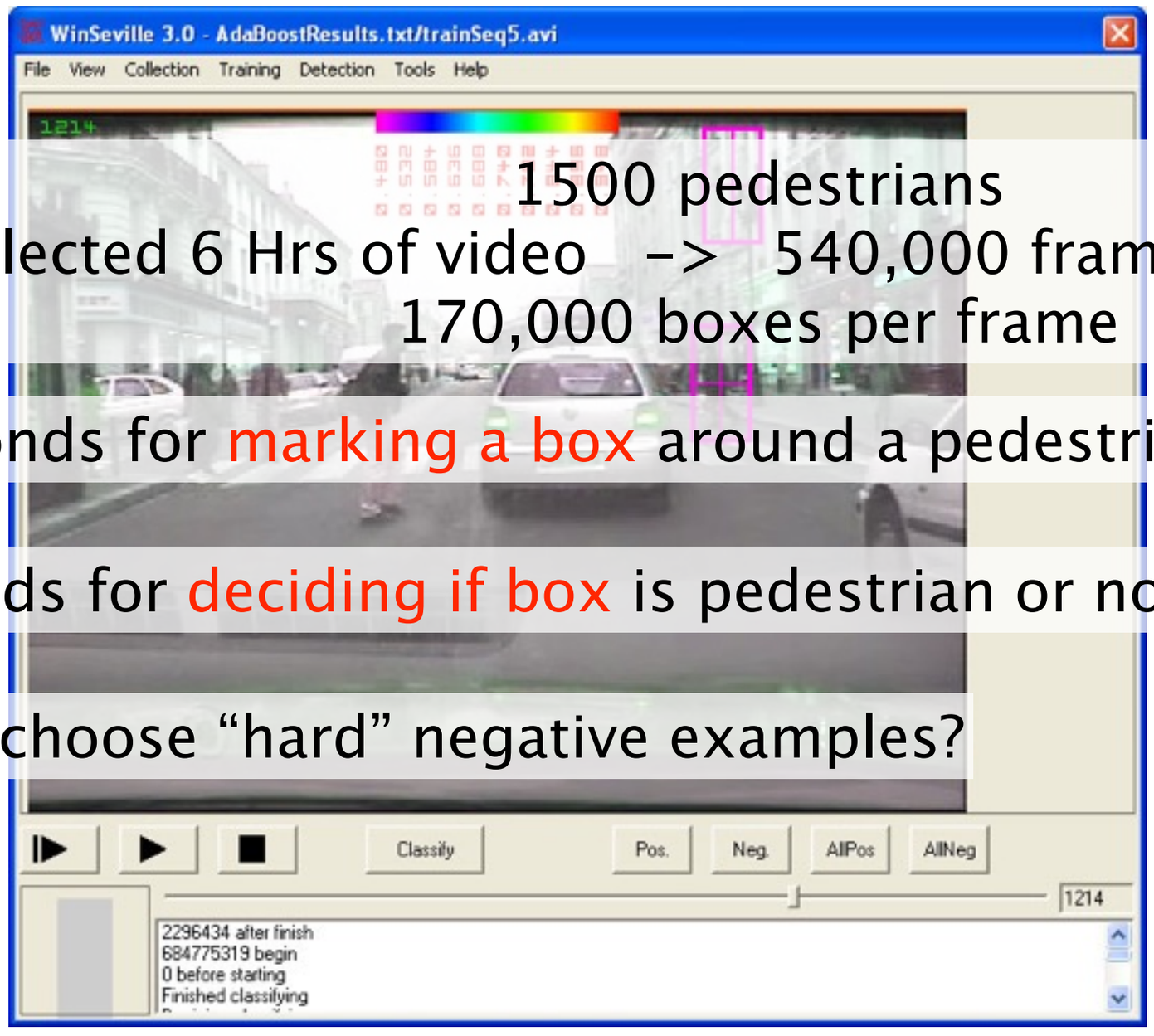
After co-training

The Seville project

- Pedestrian Alert System
- Camera mounted on front of car.
- Funded by Renault
- Collaboration with Yotam Abramson (Then at Ecole Des Mines, Paris).

Pedestrian detection - typical segment

The training process



1500 pedestrians

Collected 6 Hrs of video -> 540,000 frames

170,000 boxes per frame

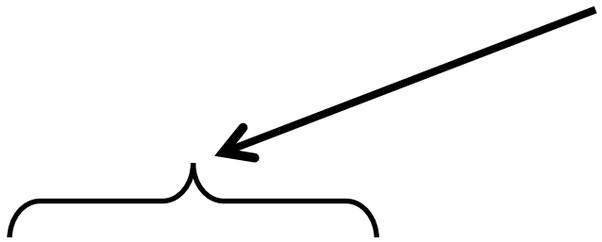
20 seconds for marking a box around a pedestrian.

3 seconds for deciding if box is pedestrian or not.

How to choose “hard” negative examples?

summary of active training

Only examples whose normalized score is in this range are hand – labeled

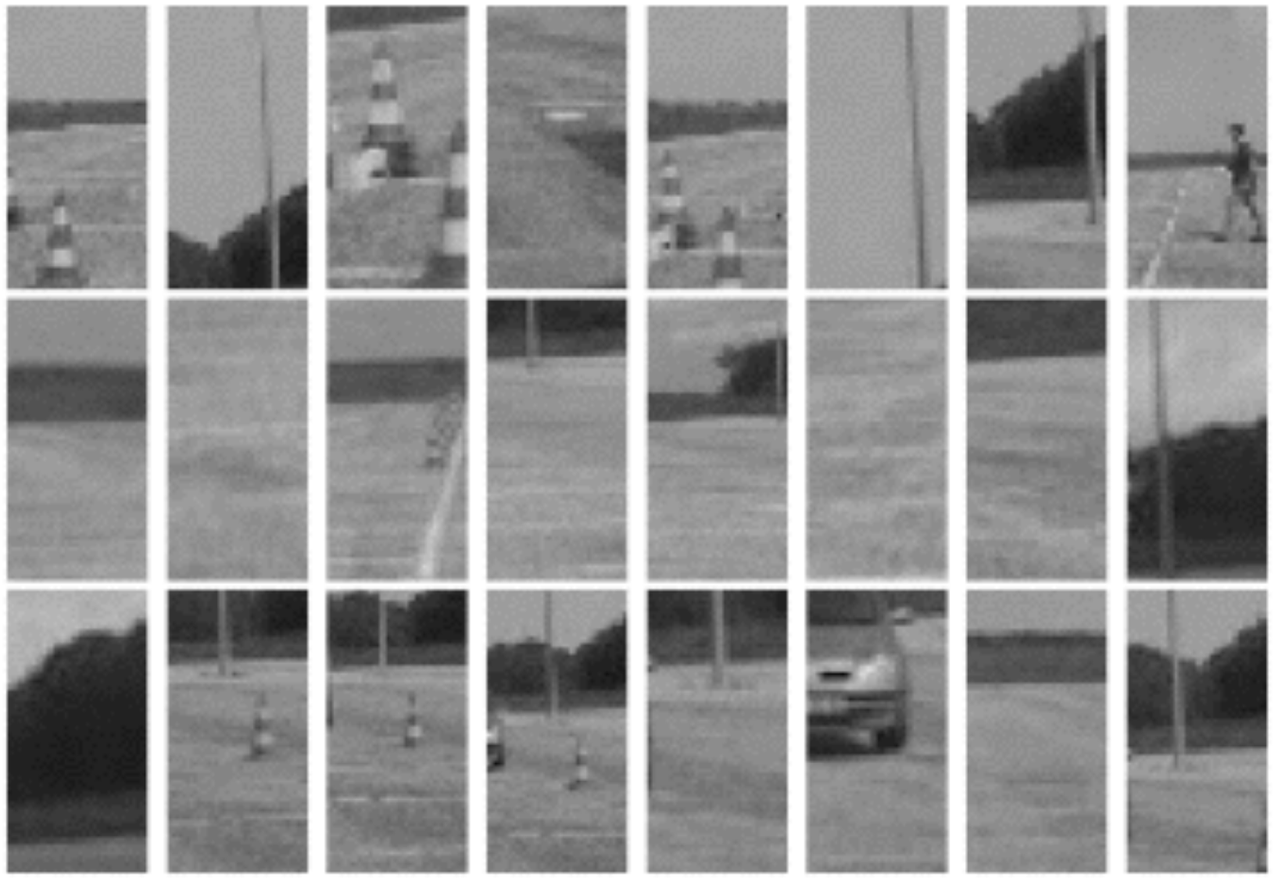


Step	total candidates	μ^-	μ^+	presented	labeled	human labor	positive	negative	training time	Weak rules
1	510 K	-	-	-	16	3m	6	10	2s	1
2	680 K	0	1	364	403	3m	36	374	6s	3
3	3,400 K	0.6	1	153	156	4m	46	520	22s	7
4	66,470 K	0.4	1	805	852	10m	86	1332	1m30s	30
5	37,910 K	0.1	0.8	1350	1439	10m	182	2675	8m	59
6	116,960 K	0	0.6	5150	5364	1h30m	417	7804	1h10m	270
7	24,140 K	-0.02	0.5	1320	863	3h	848	8236	7h30m	893
8	189,550 K	-0.02	0.5	8690	8707	3h	1178	16613	17h	1500
9	209,610 K	-0.02	0.5	2933	2933	3h	1486	19238	30h	2034
10	274,210 K	-0.02	0.5	3861	3861	4h	2046	22533	30h	3150

Easy examples

Positive

Negative



Harder examples

Positive

Negative

very hard examples

Positive

Negative

Iteration

7

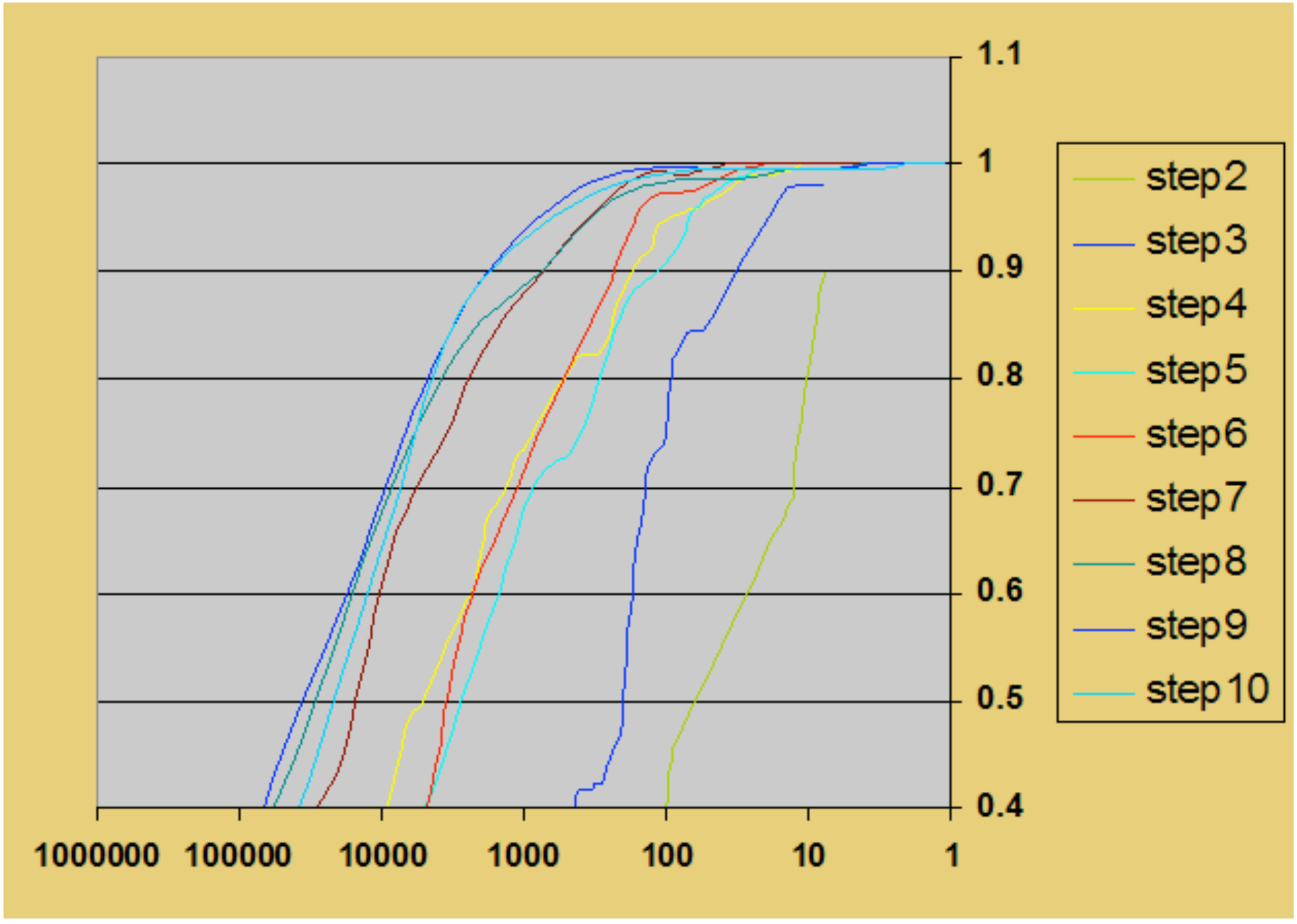
8

9

10

And the figure in the gown is ...

Detection Accuracy



Current best results

Online Boosting and Tracking

Online Boosting

[Oza & Russel 2001]

- Large data stream.
- Distribution of data changes over time.
- Partition stream into batches
  - Re-weight examples in batch using current strong learner.
  - Learn a one new weak learner.
  - Remove oldest weak learner.

Tracking using online boosting

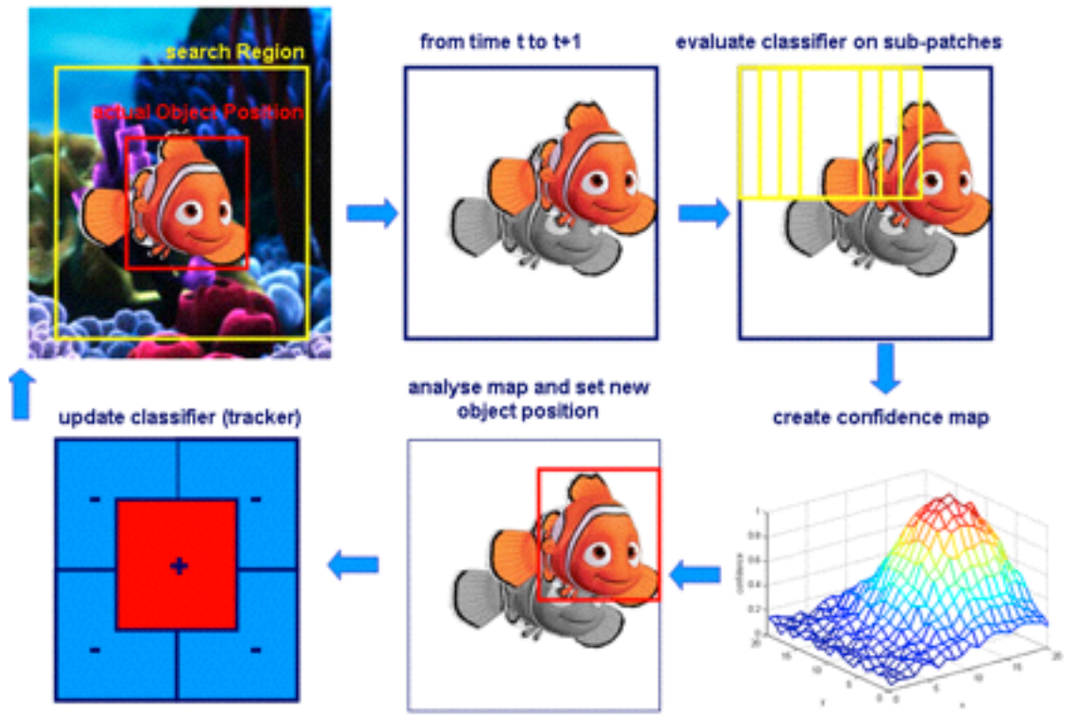
[Grabner, Grabner & Bischof 2006]

- Detect: Find tile that best fits

1. Appearance model of tracked object.
2. Constraints on movement.

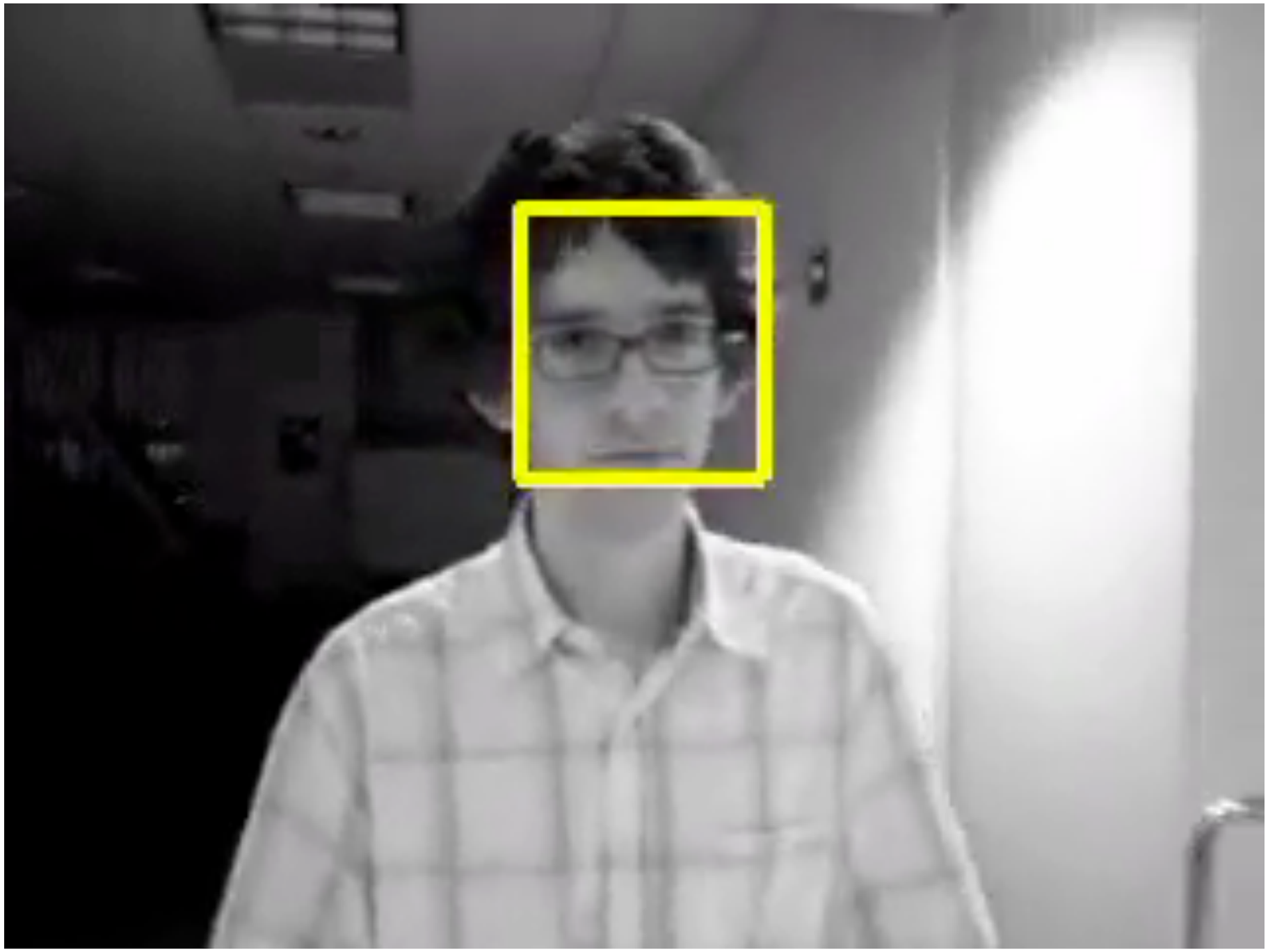
- Label: Use detected tile as positive, far tiles as negative.

- Learn: Update model using online boosting.



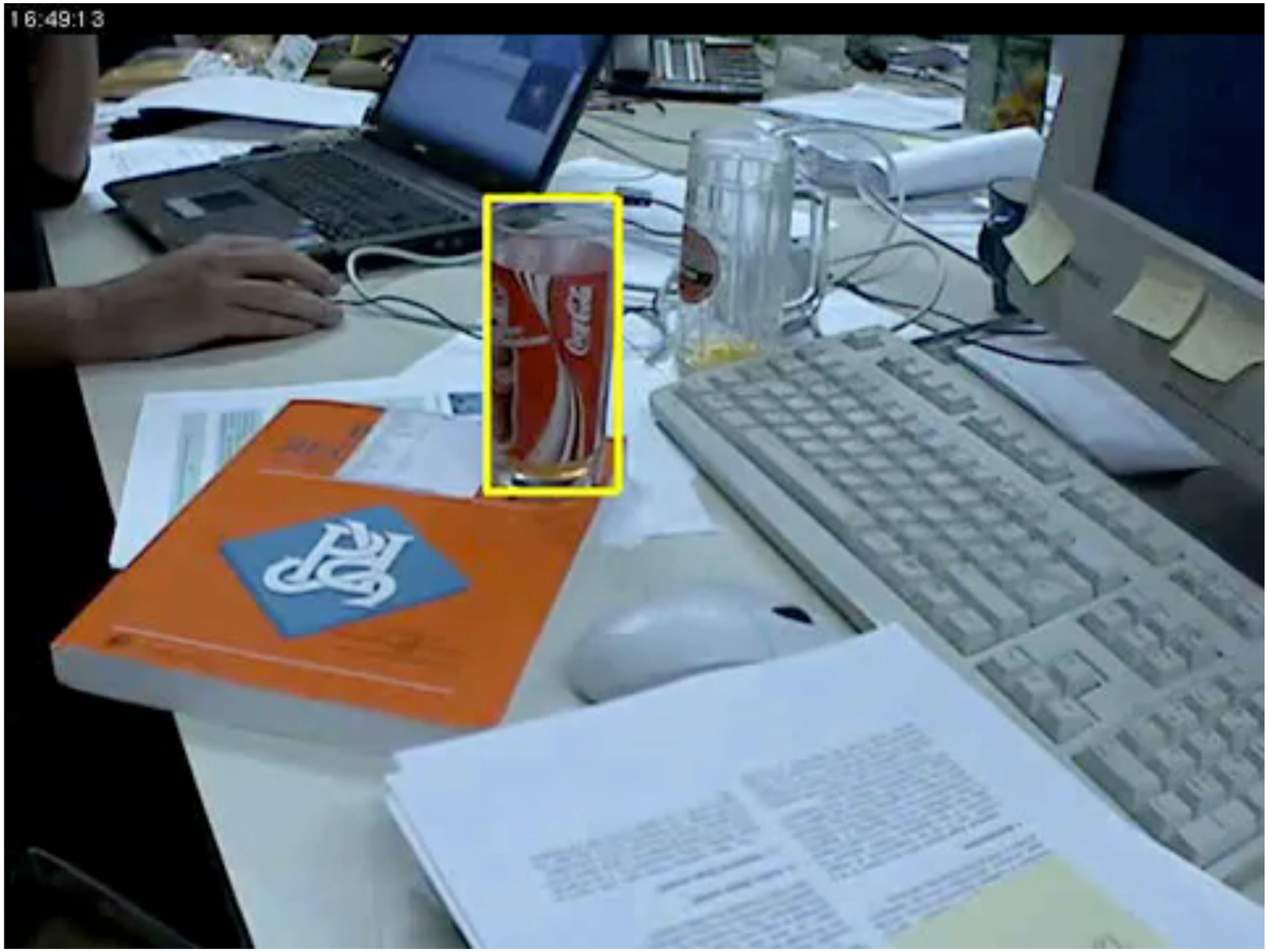
Tracking David

[Stalder & Grabner 2009]



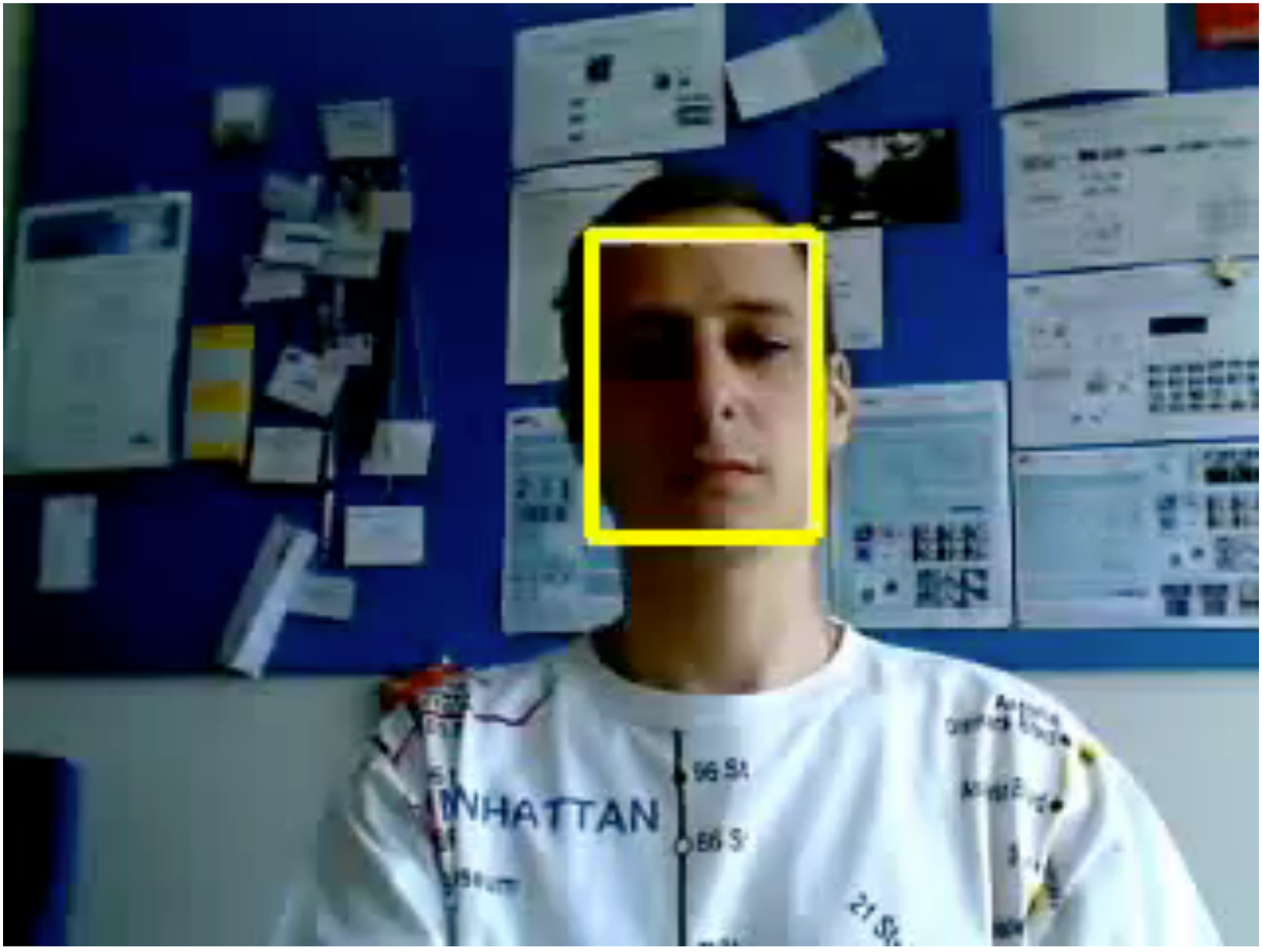
Tracking under Partial Occlusion

[Stalder & Grabner 2009]

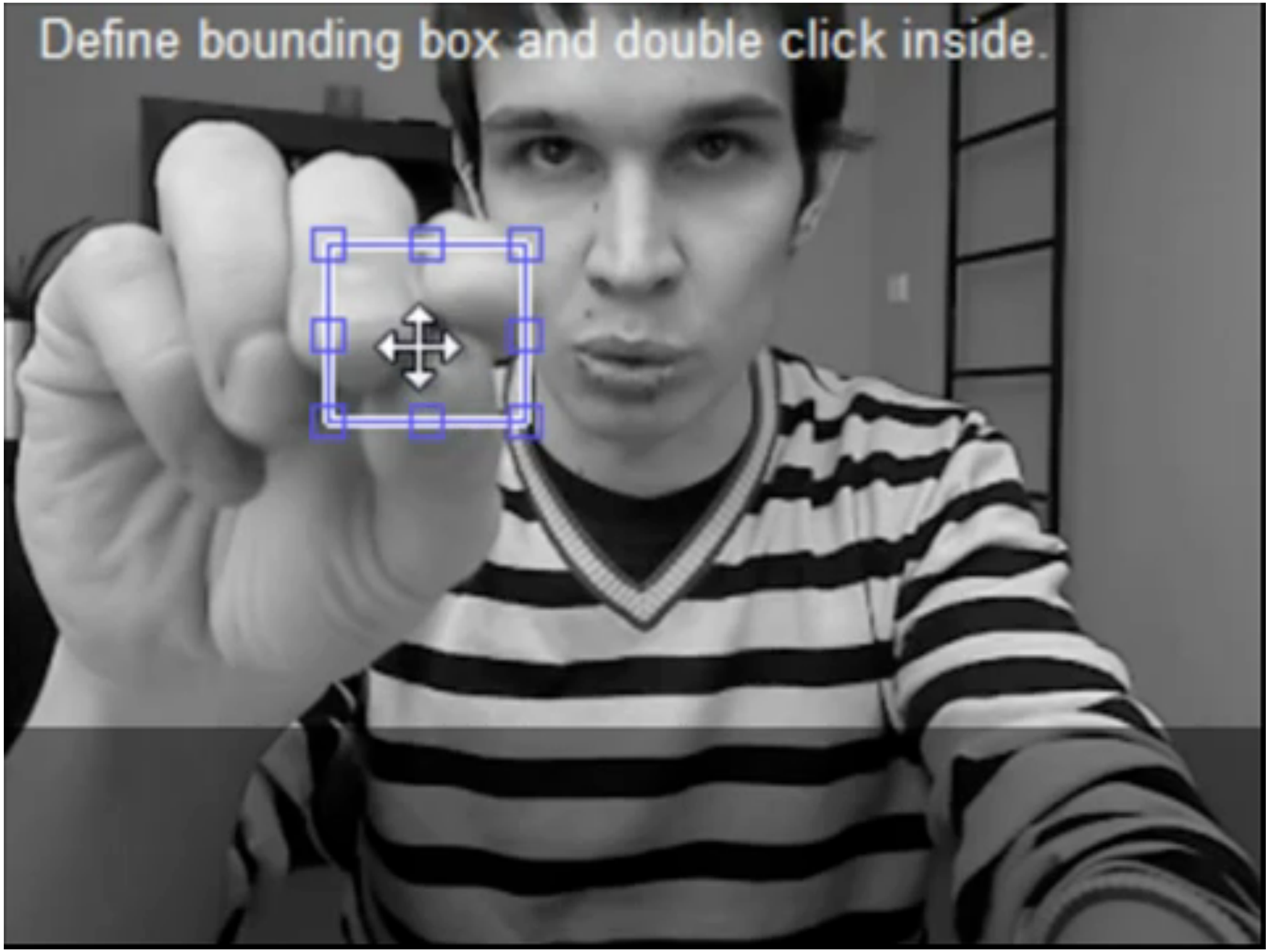


Tricking the online tracker

[Stalder & Grabner 2009]



TLD: Track, Learn, Detect



JBoost

Installation

- Go to jboost.sourcefourge.net
- Download and unzip jboost-x.x (current latest 2.3)
- Move jboost-x.x directory to a good place in your directory structure
- open a terminal and cd to the jboost-x.x directory.

Required software packages

- Needed packages:
  - java (version 1.6 works) - Base language
  - python (version 2.7.2 works) - Scripting Language
  - jboost (Latest version is 2.3)
  - GraphViz - node-edge graph visualization (2.28 works)
  - gnuplot - X-Y graph visualization (4.2 works)
  - Cygwin - a unix-like shell for Windows.

Check Versions

$ scripts/checkVersions.sh

---------- java

java version "1.6.0_33"

Java(TM) SE Runtime Environment (build 1.6.0_33-b03-424-10M3720)

Java HotSpot(TM) 64-Bit Server VM (build 20.8-b03-424, mixed mode)

---------- python

Python 2.7.2

---------- gnuplot

gnuplot 4.2 patchlevel 5

---------- graphviz

dot - graphviz version 2.28.0 (20110509.1545)

Quick Start

- After installation and checking versions perform:
  - source setPath.sh
  - scripts/runScripts.sh
  - Lets switch to a terminal

Genome-Wide Association Studies

Genetic Disorders

- The influence of heredity on disease.
- Mendalian Diseases: Influenced by a single gene:
    - Sickle-cell Anemia - two copies of a single recessive gene.
        - One copy increases resistance to Malaria.
- Non Mendalian diseases are influenced by many genes.

GWAS, the idea

- According to longitudinal studies many common diseases have a significant heritable component.
    - High Blood Pressure, Diabetes, Cron Disease, Otism ...
- Can we find which genes are the culprits?
    - Genome Wide Association Studies: sequence ~500,000 DNA locations (SNPs) on patients (and controls)
    - Use statistical methods to find associations (correlations) between DNA location and disease.

GWAS, current status

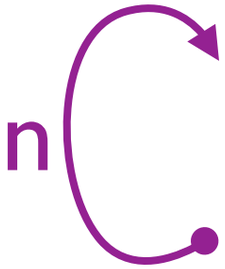
- Several large datasets (5,000 - 10,000) published (but getting access is not trivial)
- Association studies find a few SNPs with statistically significant correlation. But,
- The percentage of variance explained is usually low (1% - 5%)
    - Especially glaring for universal traits such as height.

Machine learning to the rescue!

- Instead of finding correlations between disease and single SNPs, learn a function that maps the SNP vector to the disease.
- Find the set of SNPs on which the function depends.
- Good idea, people did it using SVM, random forests, ...
- Good test set performance
  - BUT: the geneticists are not convinced.
- Predictability does not imply causality.
- What is the p-value?

Boost-Remove

- We have 500,000 features (SNPs)
  - Run Boosting for k (50) iterations.
  - Remove the SNPs used.
- Consider all of nxk SNPs



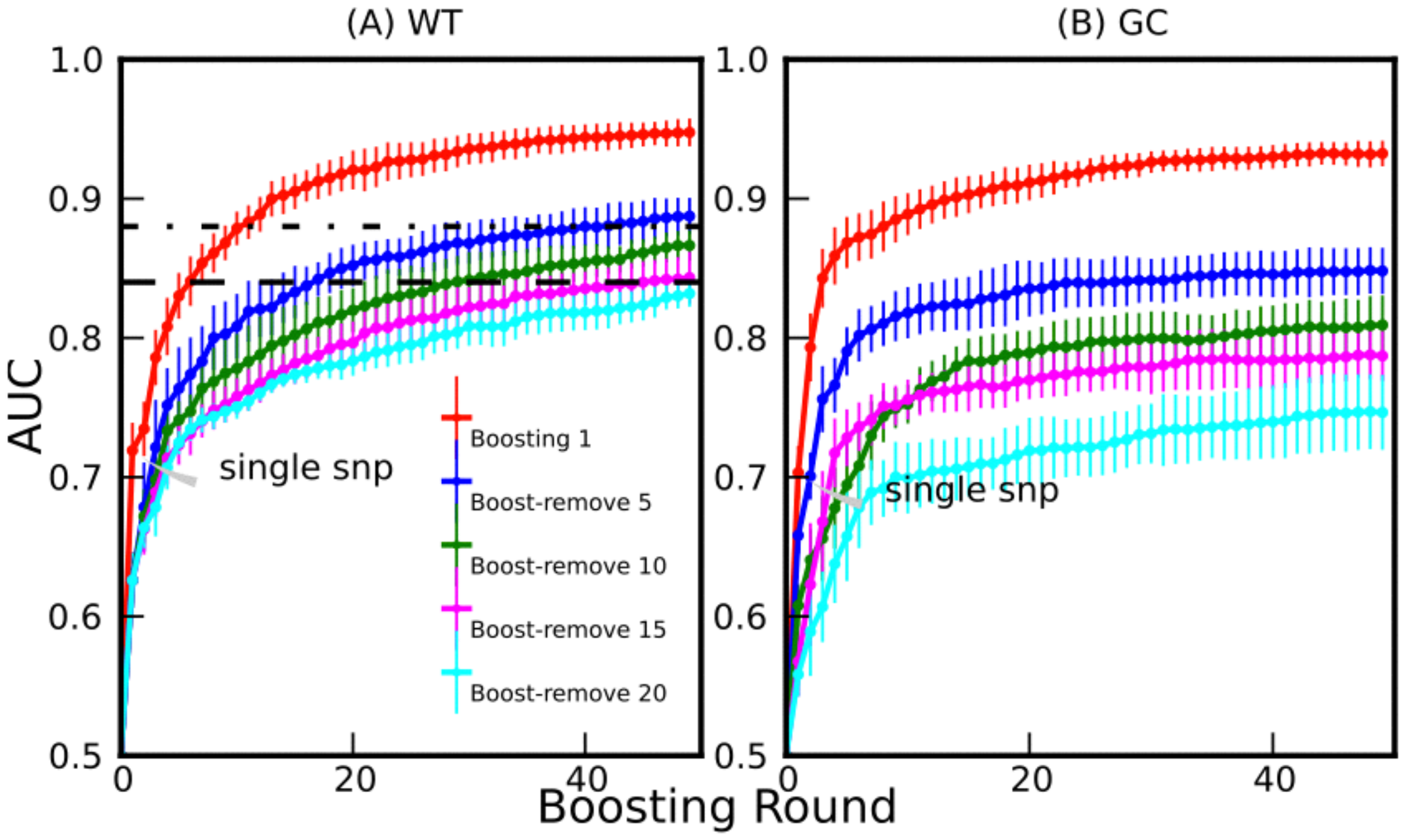
Why is it hard to interpret?

- Linkage Disequilibrium: dependencies between SNPs:
  - Location Linkage: recombination rate depends on distance btwn SNPs.
  - Population Stratification: groups of related people (ethnicities)
  - Selection: Fitness depends on combination of SNP states.
  - Different mutation rates, selective mating ...
- Result: many non-causal correlations.

Results on two datasets

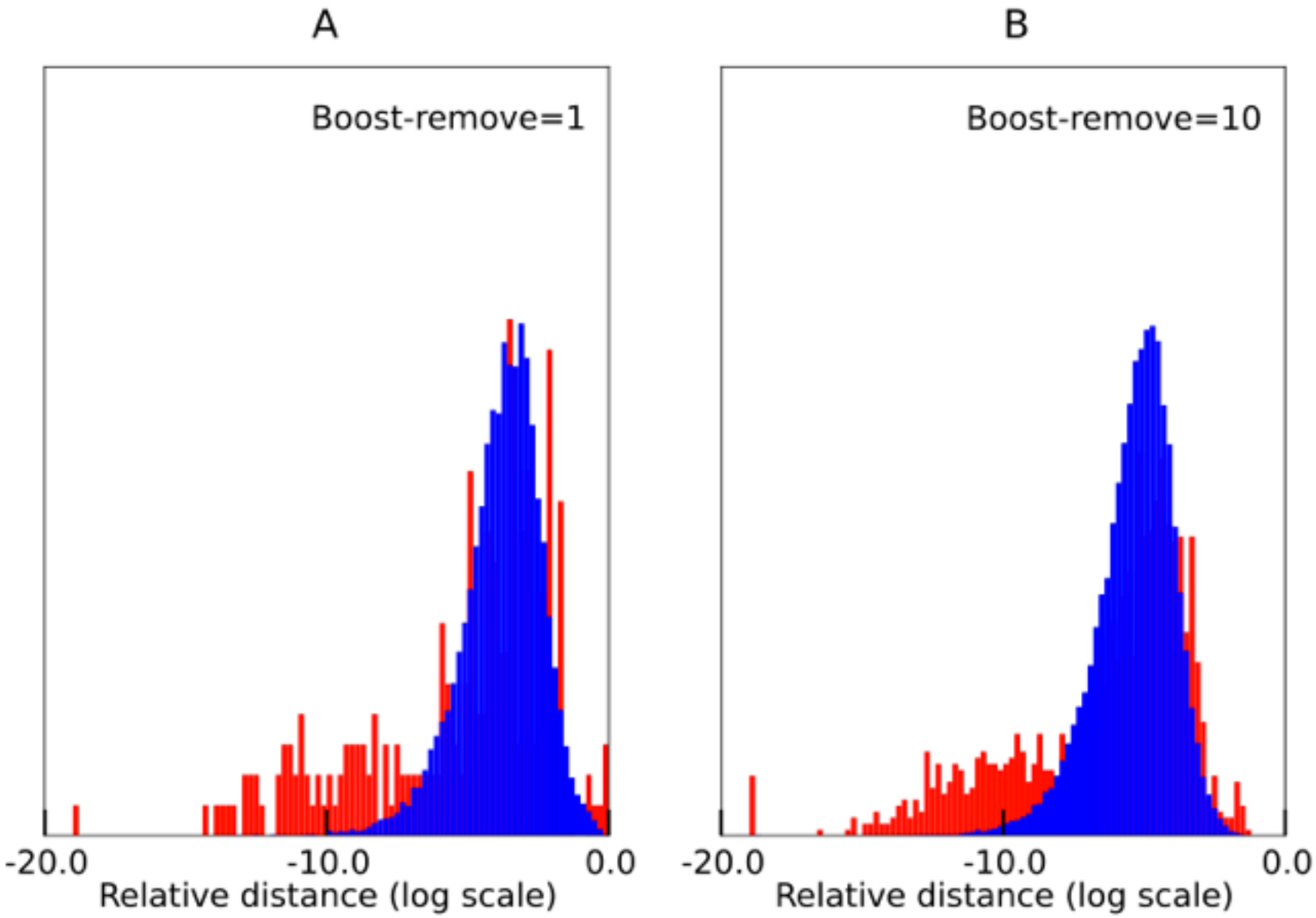
WT consortium: 2000 cases, 3000 controls

GC consortium: 4061 cases and 2571 controls



Measuring closeness of location

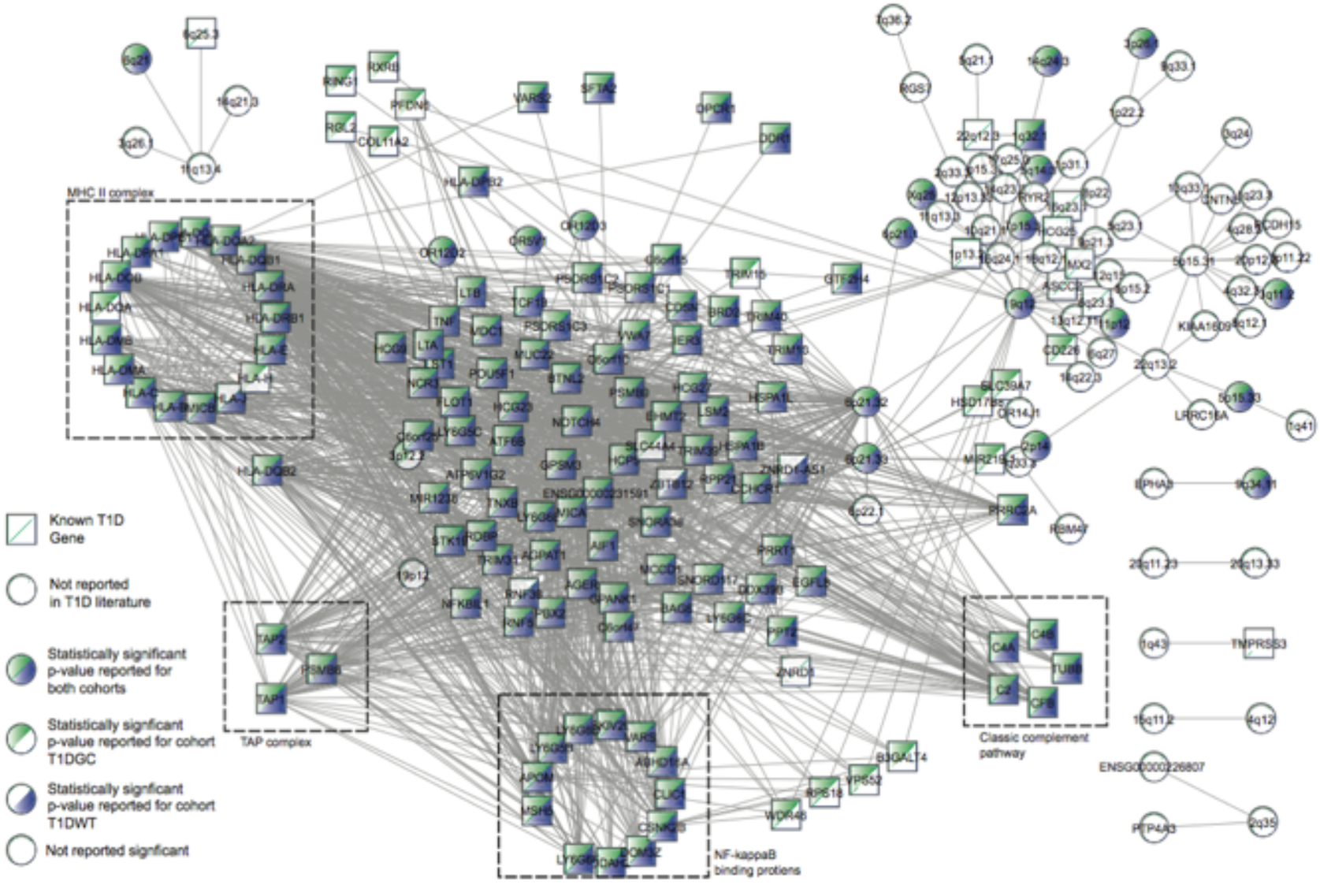
Location Consistency

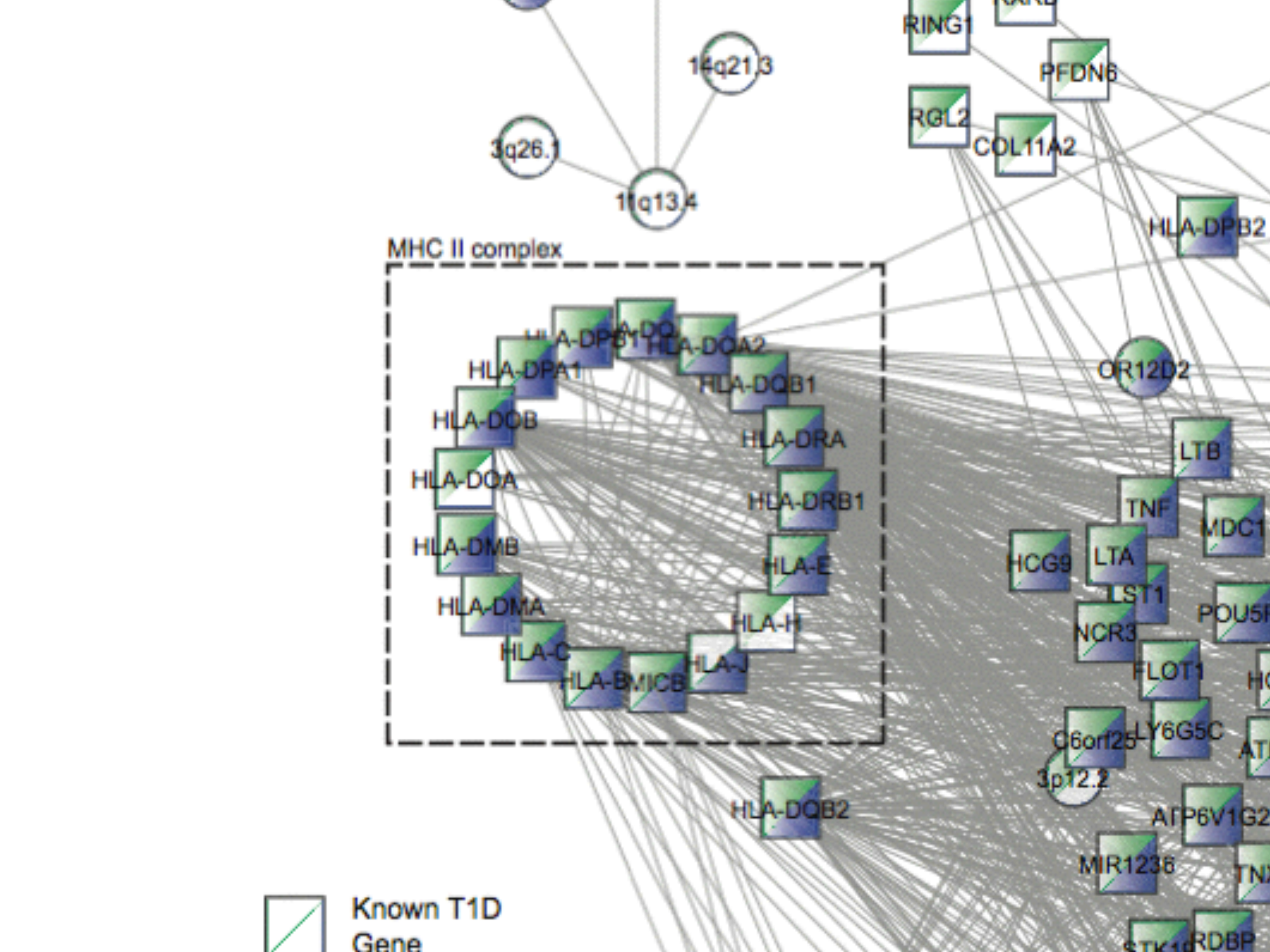


Mann-Whitney U test yields $p=10^{-30}$

related SNPs

Tree structure of ADT hints at relations btwn SNPs

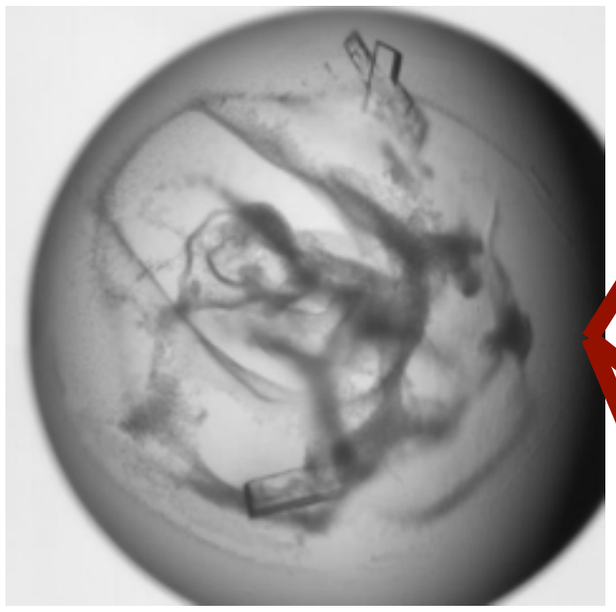


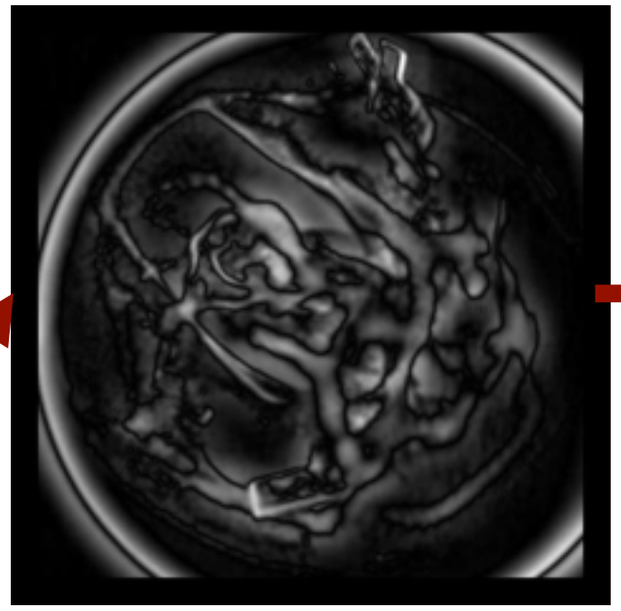

14q21.3
3q26.1
11q13.4
RING1
PFDN6
RGL2
COL11A2
HLA-DPB2
MHC II complex
HLA-DPB1
HLA-DQA2
HLA-DPA1
HLA-DQB1
HLA-DOB
HLA-DRA
HLA-DOA
HLA-DRB1
HLA-DMB
HLA-E
HLA-DMA
HLA-H
HLA-C
HLA-J
HLA-B
MICB
OR12D2
LTB
TNF
MDC1
HCG9
LTA
LST1
NCR3
FLOT1
C6orf25
LY6G5C
3p12.2
HLA-DQB2
ATP6V1G2
MIR1236
RDBP
Known T1D
Gene

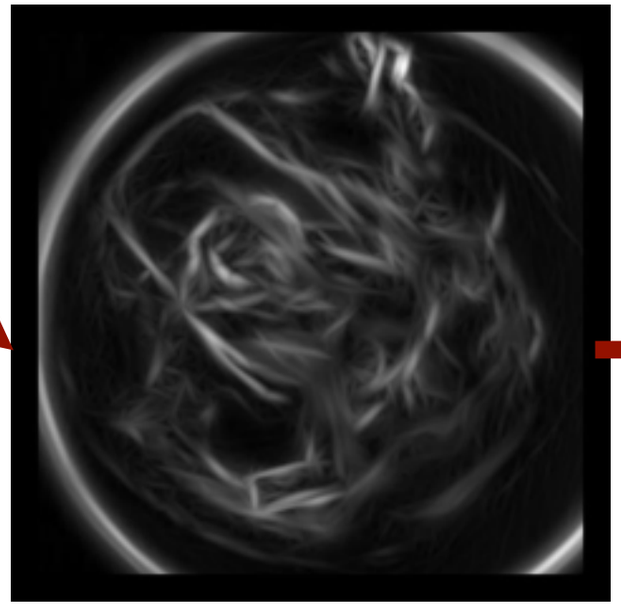
Protein Crystallography

- Best current method for finding the 3D structure of protein.
- Huge investments ($100,000,000's/year)
- Main hurdle: get proteins to form crystals.
- Highly automated pipeline
- Manual bottleneck: deciding which droplets have "harvestable crystals".
- Our goal: automate crystal detection.

Feature Extraction



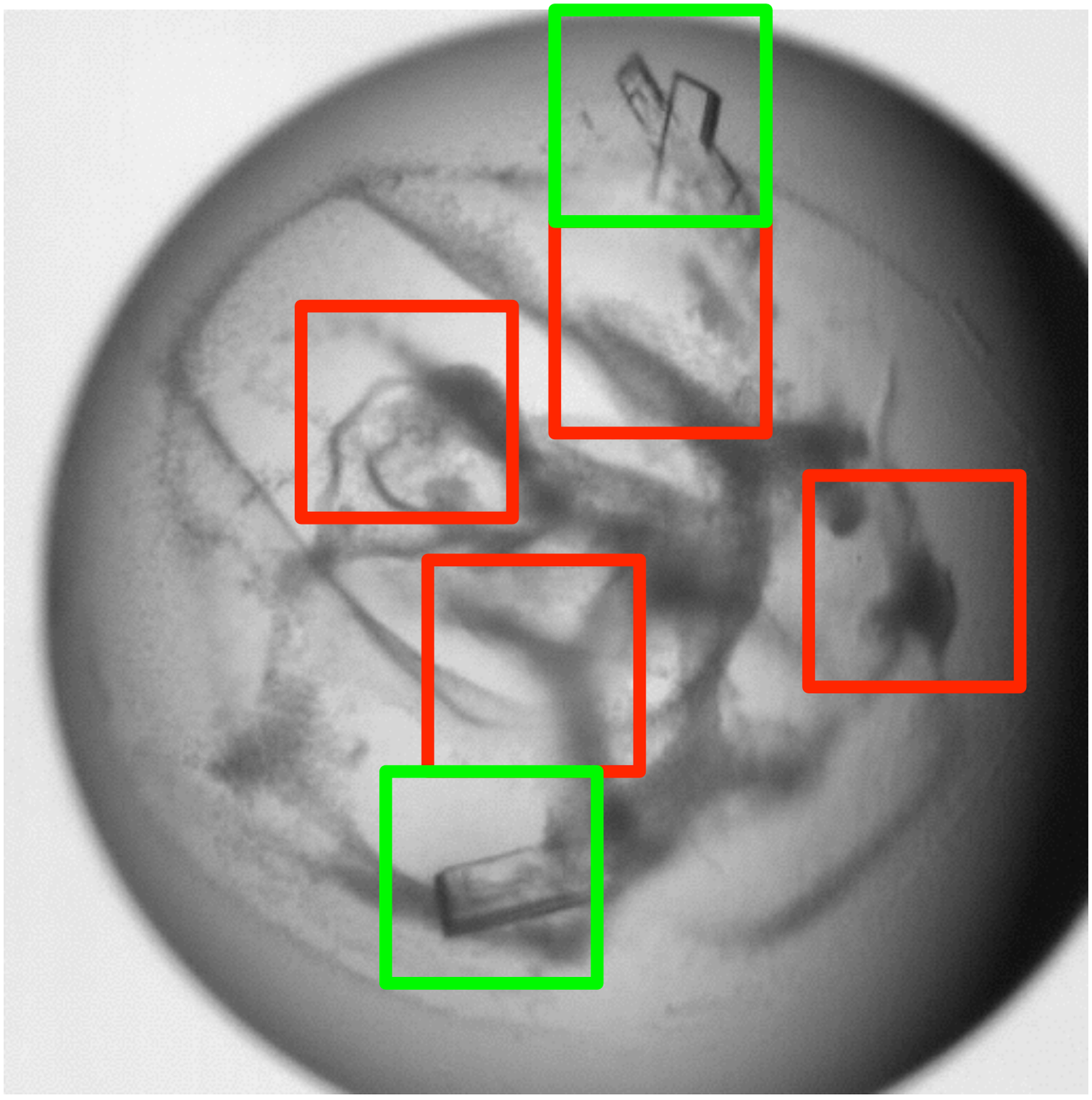




0.00005101
0.00003512
0.00001512
0.00000321
0.21335881
0.00000696
0.00002197
0.00001106

0.00047457
0.00030493
0.00013458
0.00001244
0.00000317
0.00000134
0.00000044
0.00000000

Window Scanning



Extracted features suggest precipitant

0.00064995
0.09151233
0.00178373
0.27644670
0.00250930
0.53748393
0.04090499
0.03716168

0.14000916
0.00001396
0.00005037
0.00004302
0.00003734
0.00001220
1.02424619
0.00010212

Extracted features suggest crystal hits

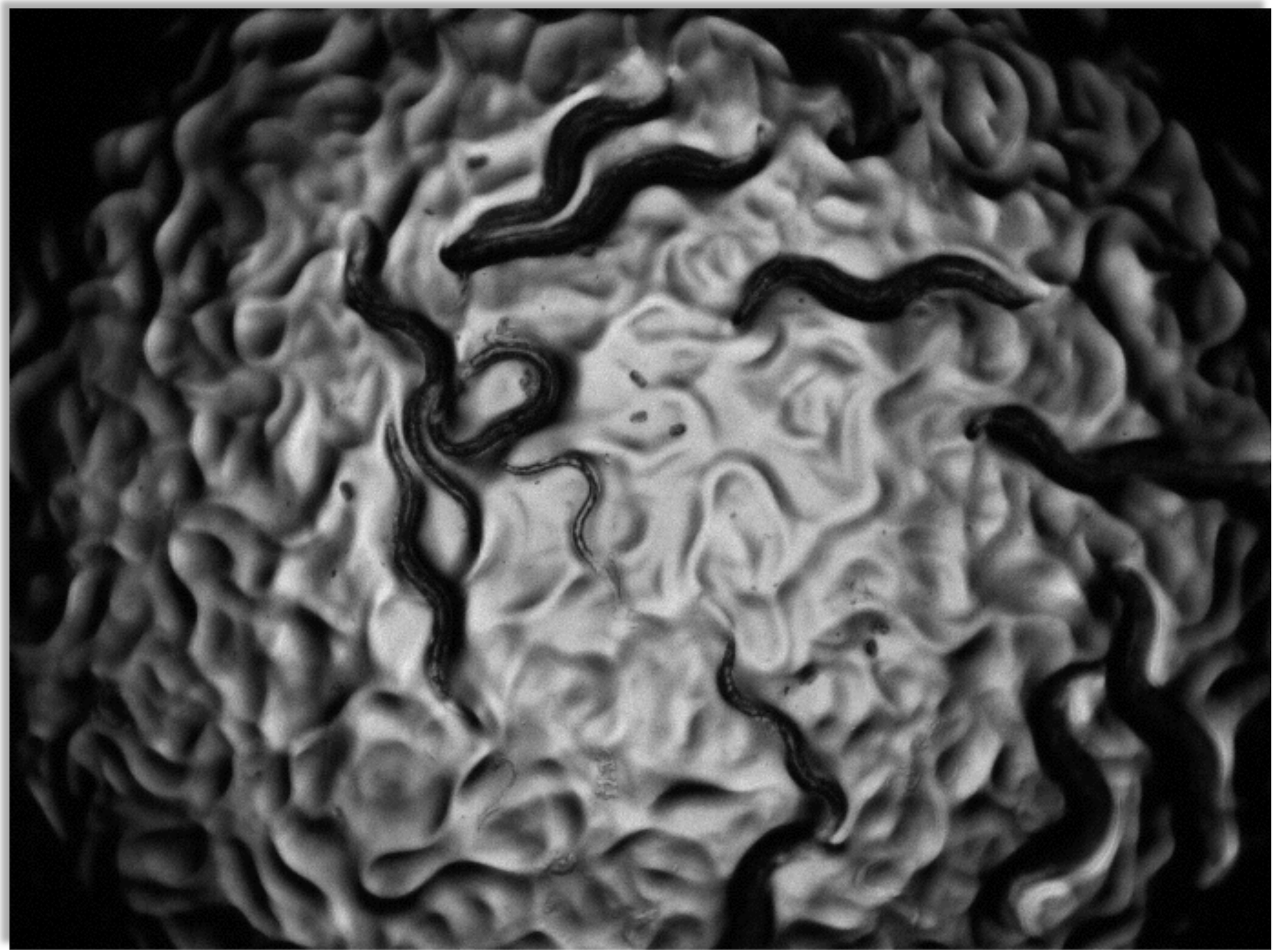
Analysis of C. Elegans

- microscopic worm is a popular model organism in biology.

- Worms are bred in pleasant medium of agar. (Pleasant for worms not for biologists.)

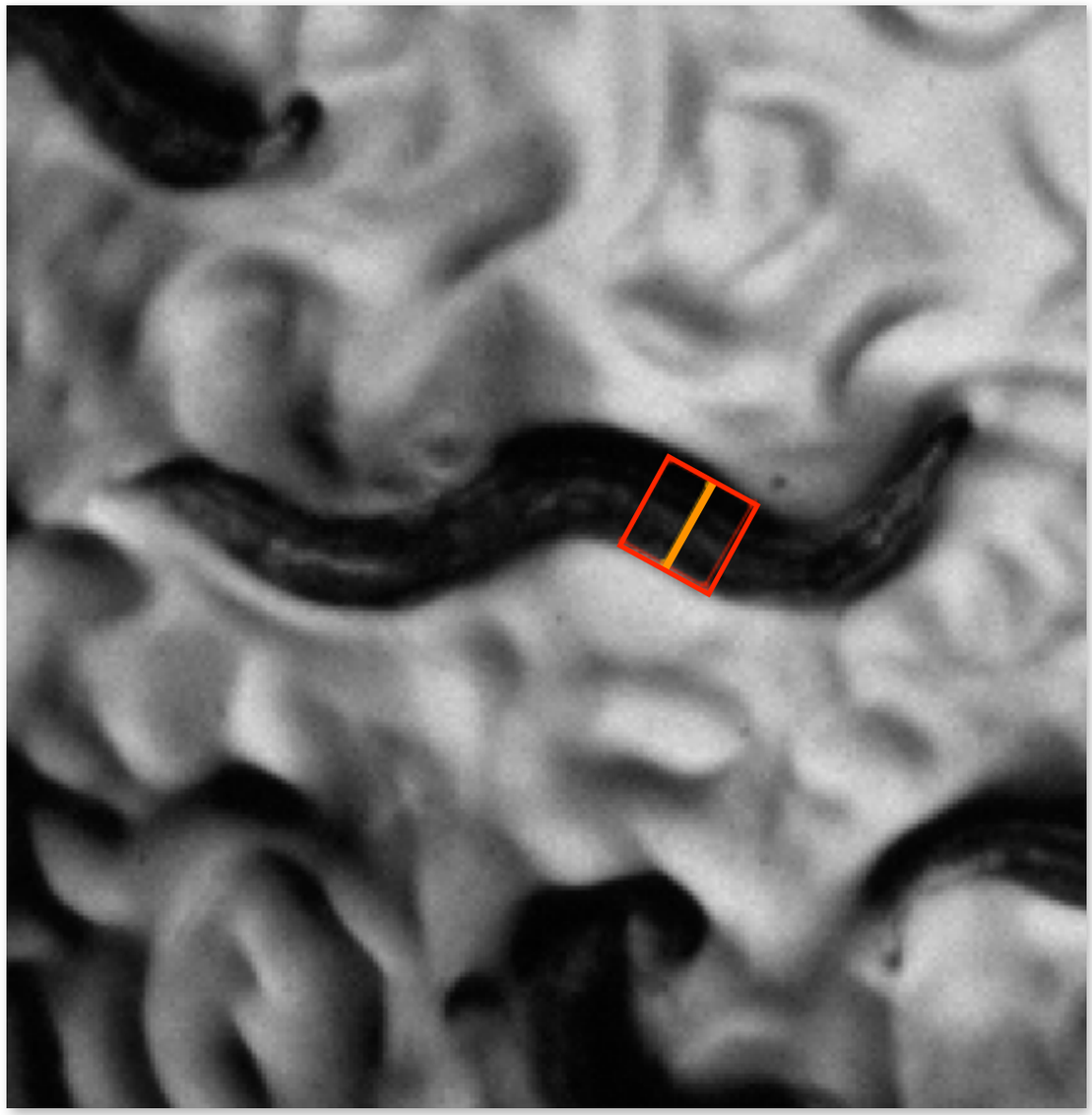
- Worms are imaged under normal light and fluorescent light.

- Collaboration with Anne Carpenter (Broad institute) and Annie Lee Connery (MGH, Ruvkun Lab and Ausubel Lab).

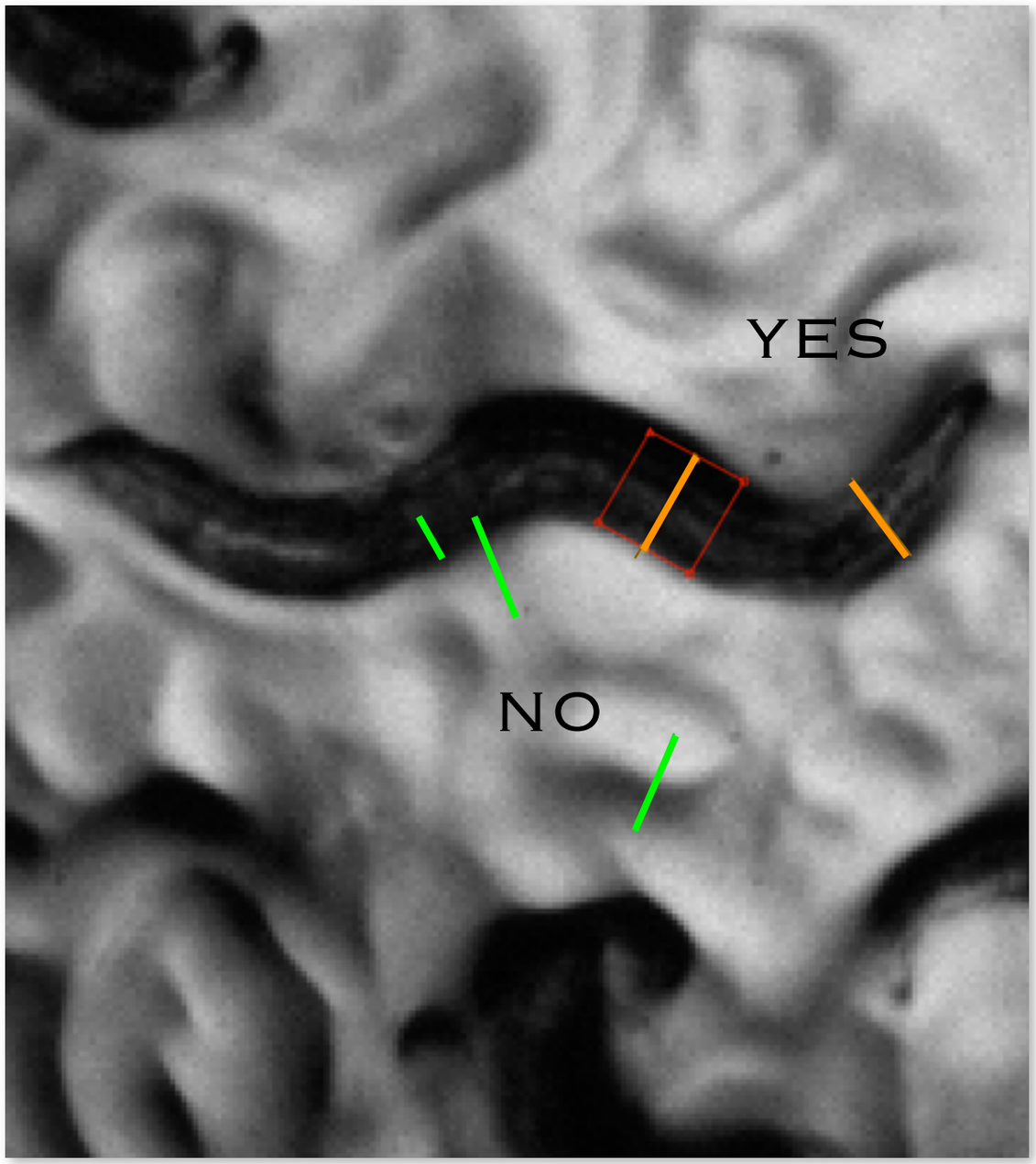


Goal: detecting worm segments

- Center line represents box.
- Identify:
  - location
  - size
  - orientation

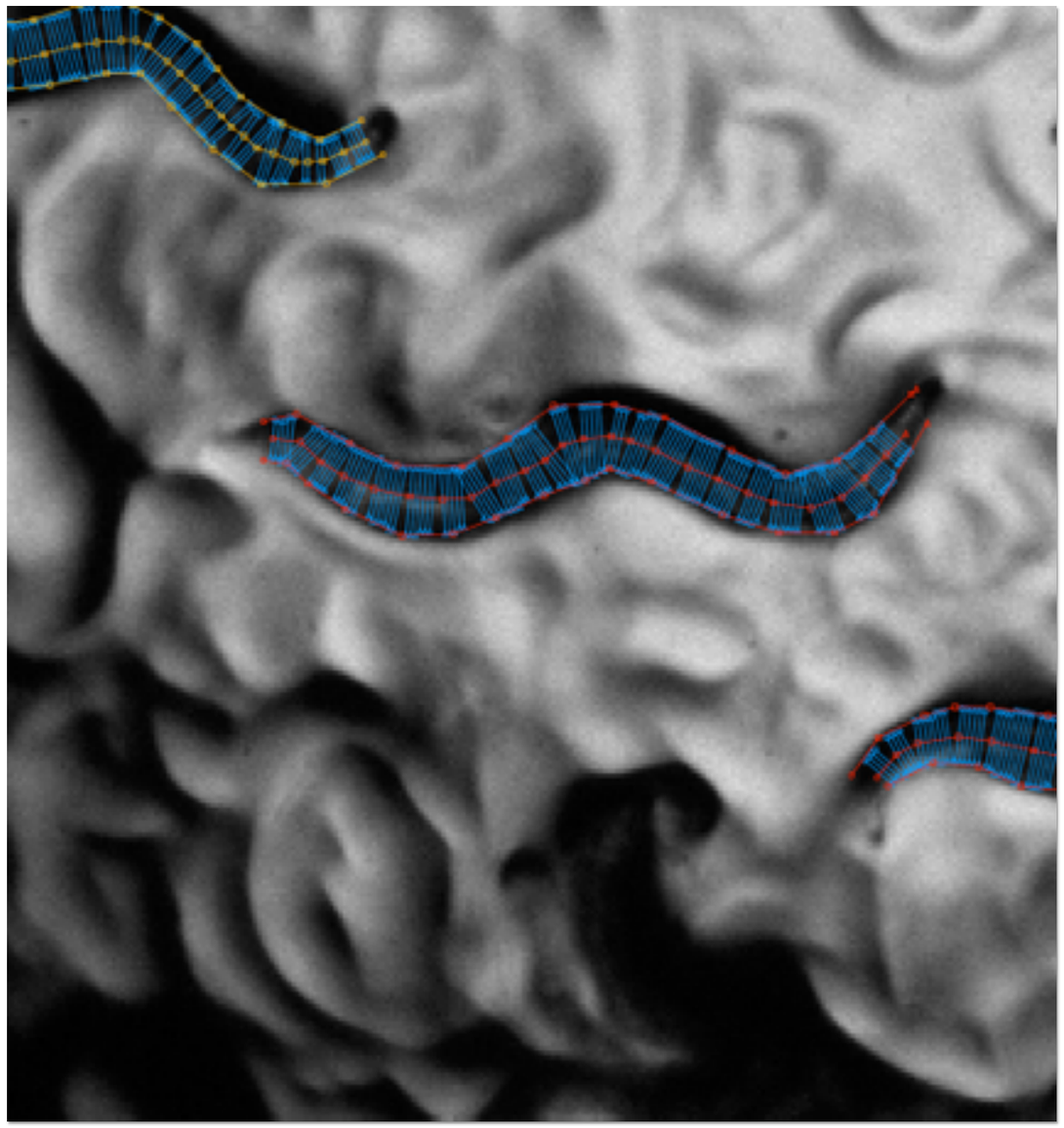


- Positive and negative training examples.

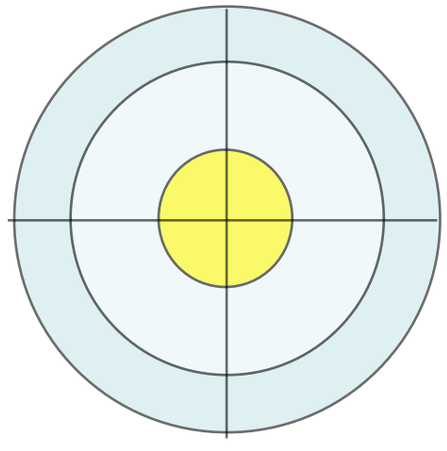
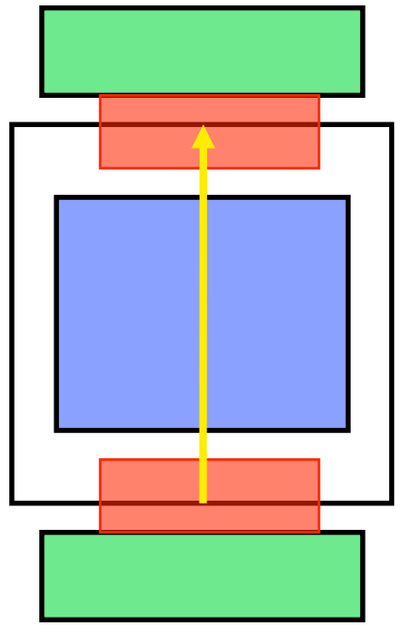


Annotation tool

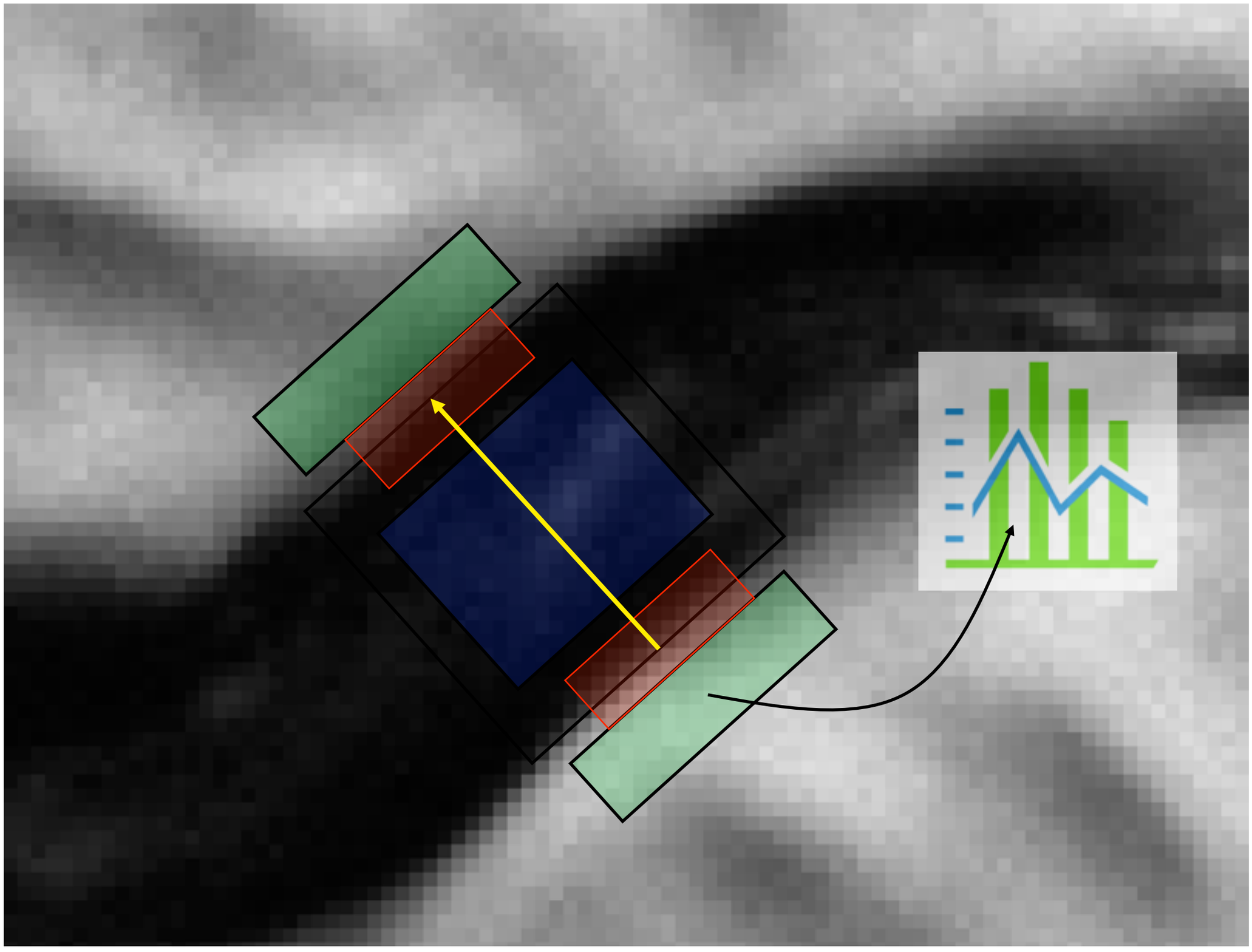
- User annotates worm boundaries.
- Computer generates positive examples.
- Negative examples chosen randomly.



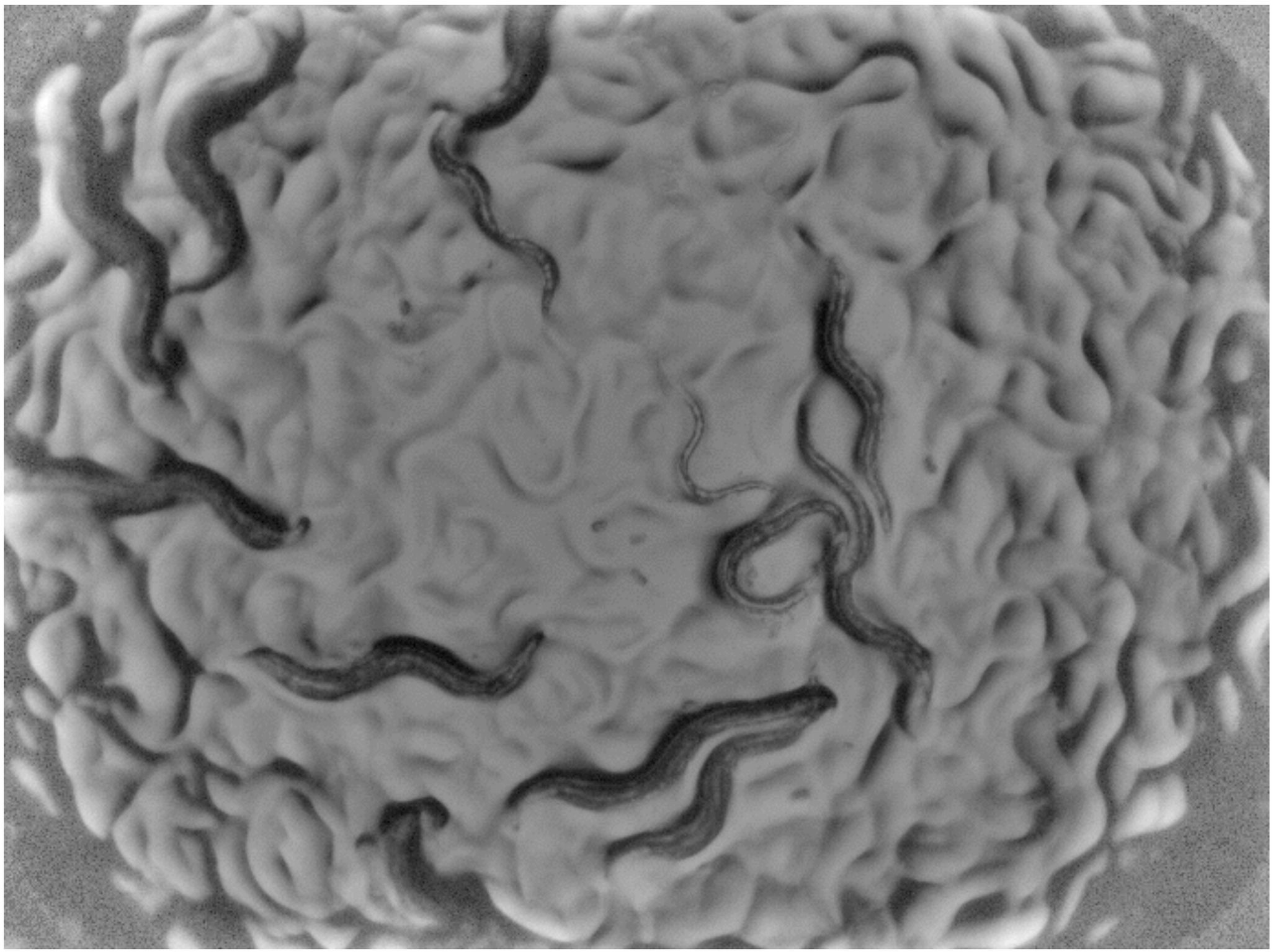
The computer vision features

	Yeast	Worm Segment
Pre-processing	Edges Bright Field	Edges, LoG Bright field, Fluorescence
Mask		

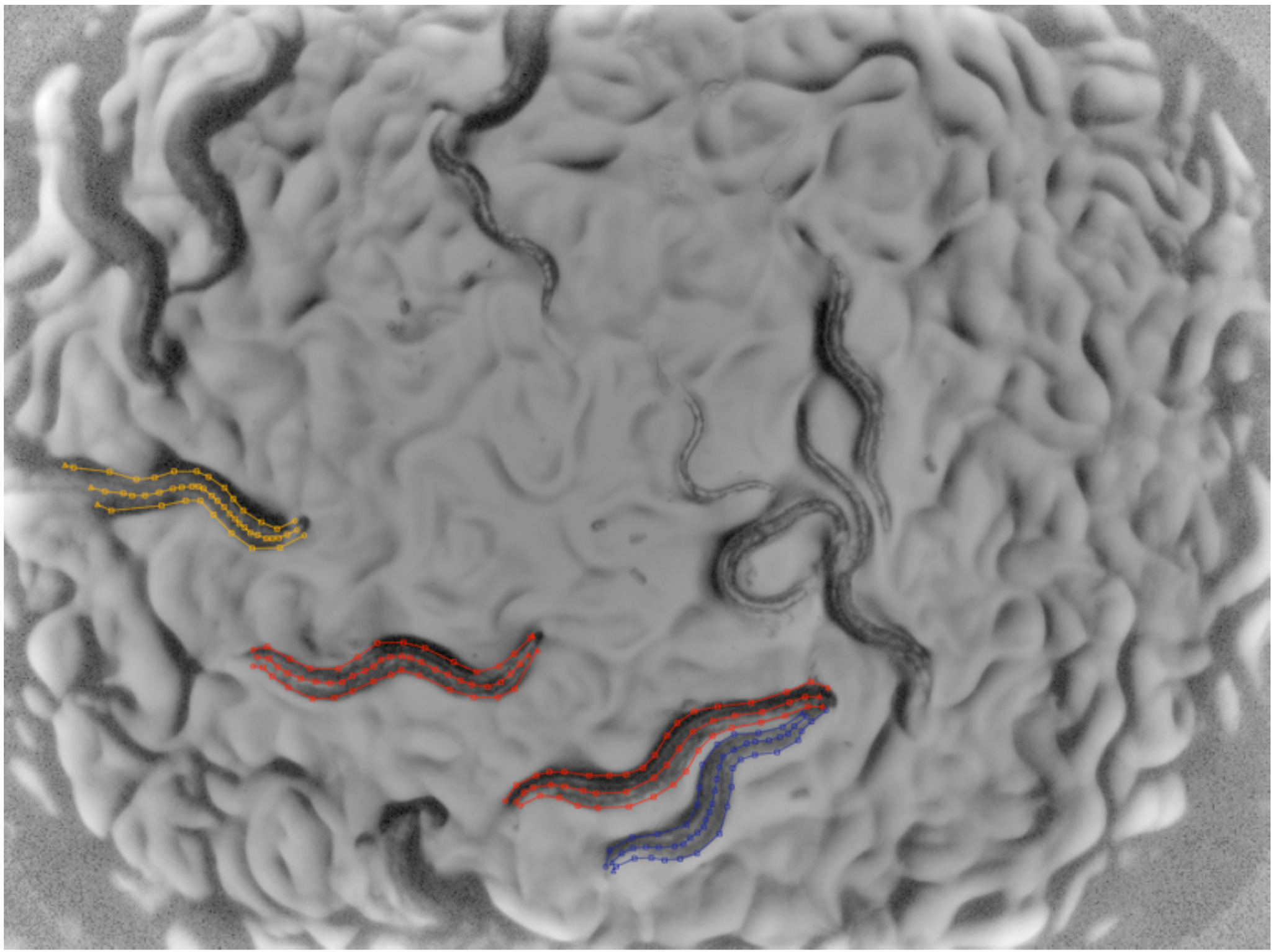
Feature Calculation



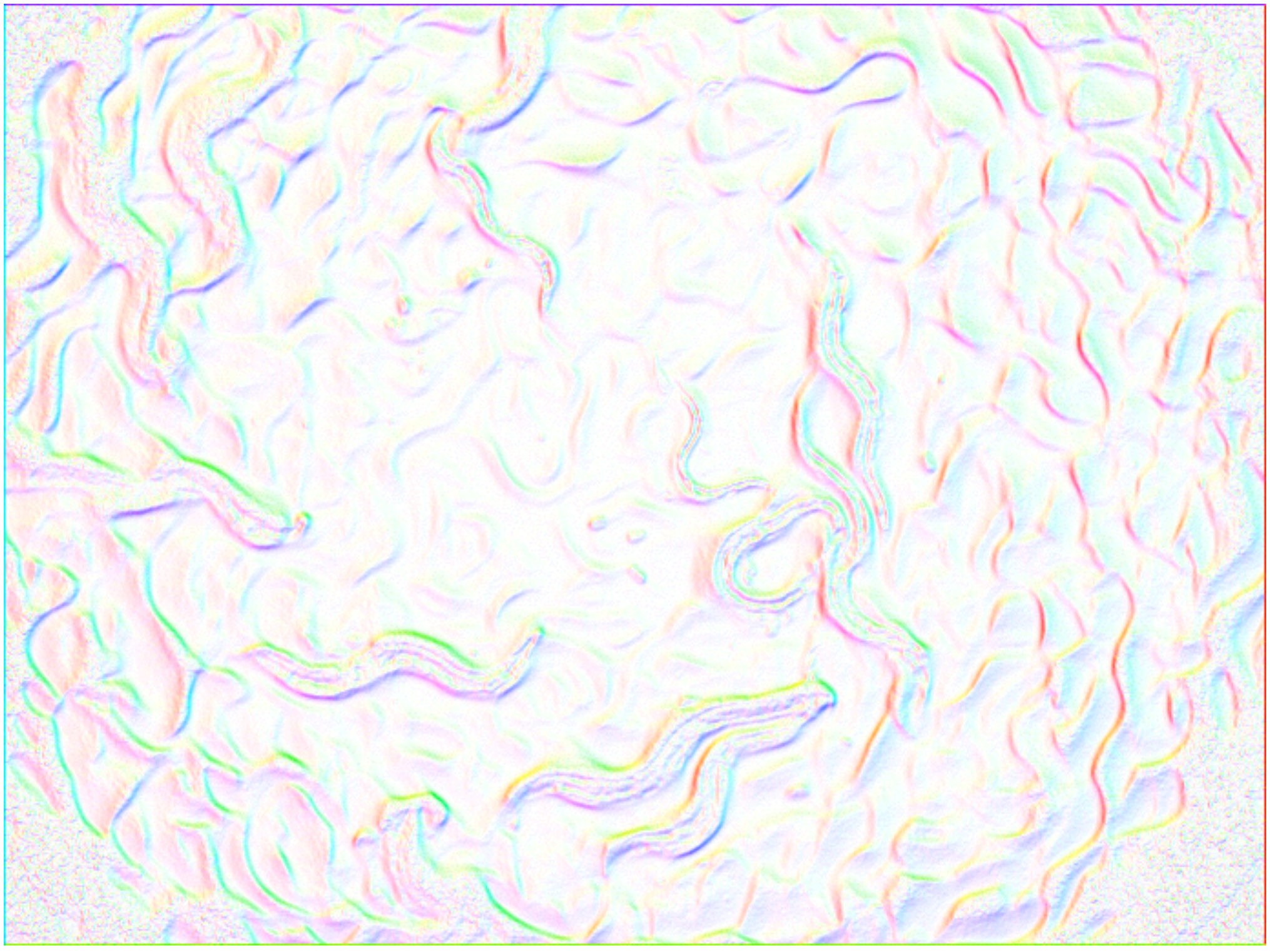
Preprocessing: Normalizing



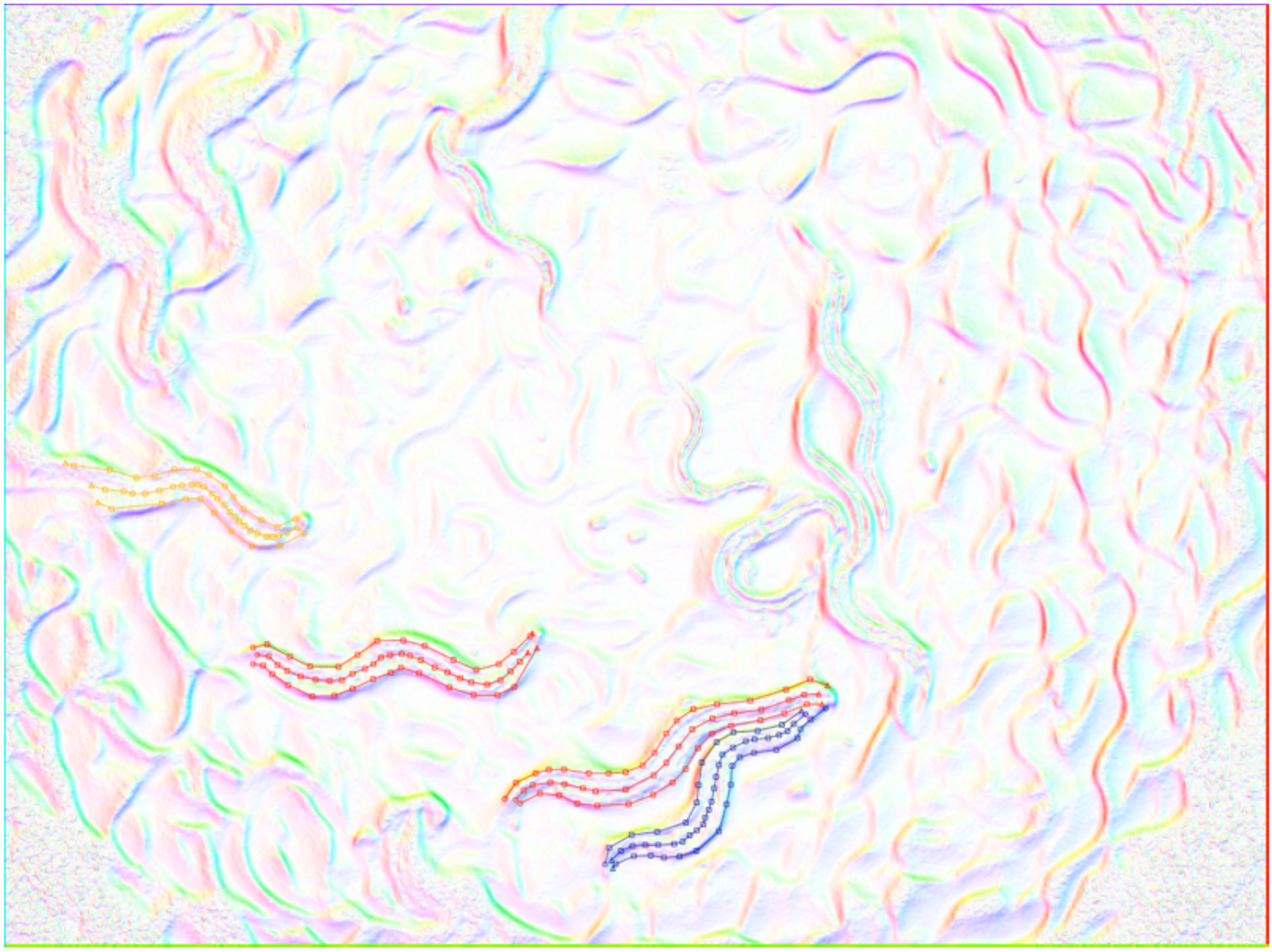
Preprocessing: Normalizing



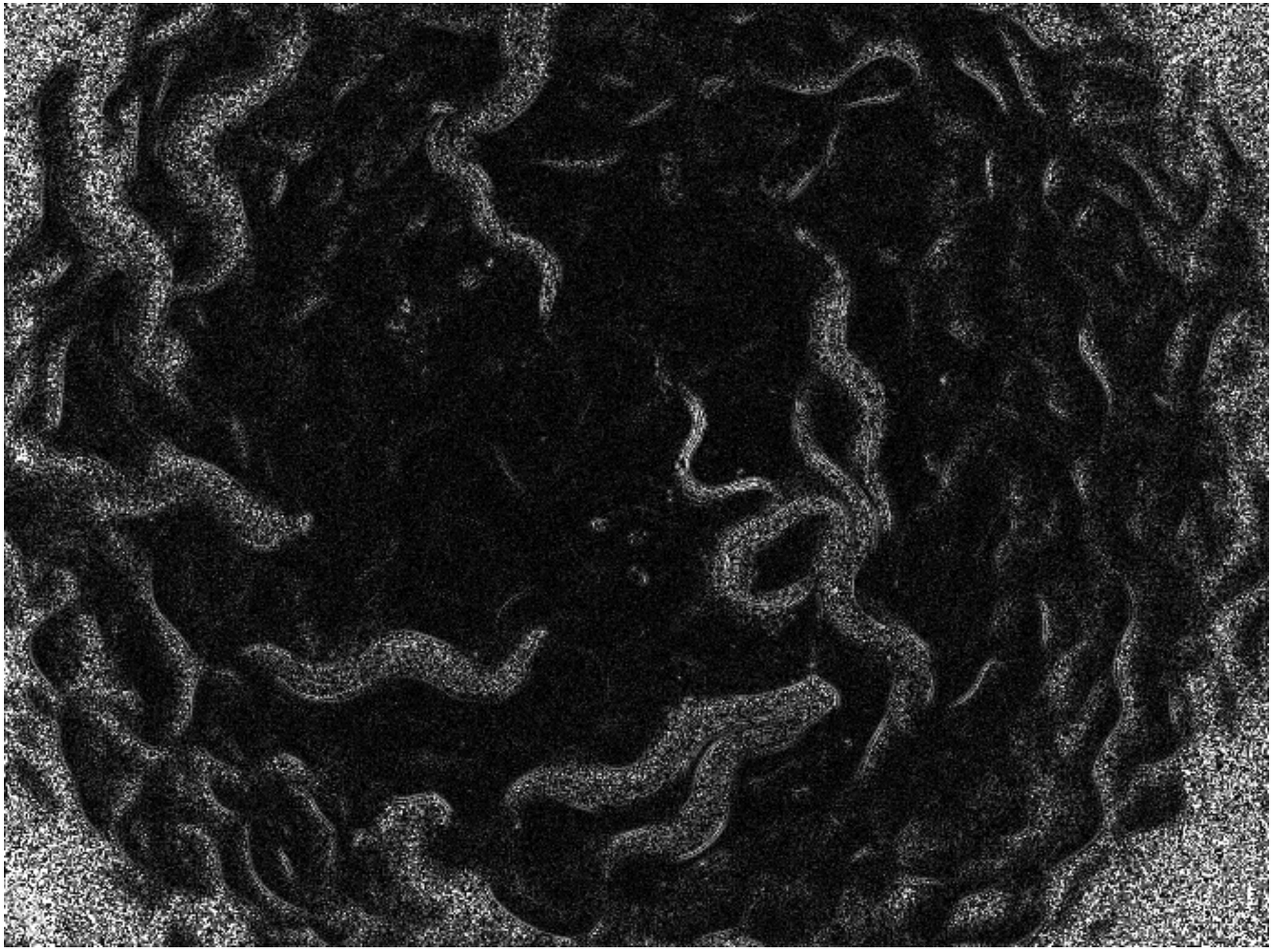
Preprocessing: Edges



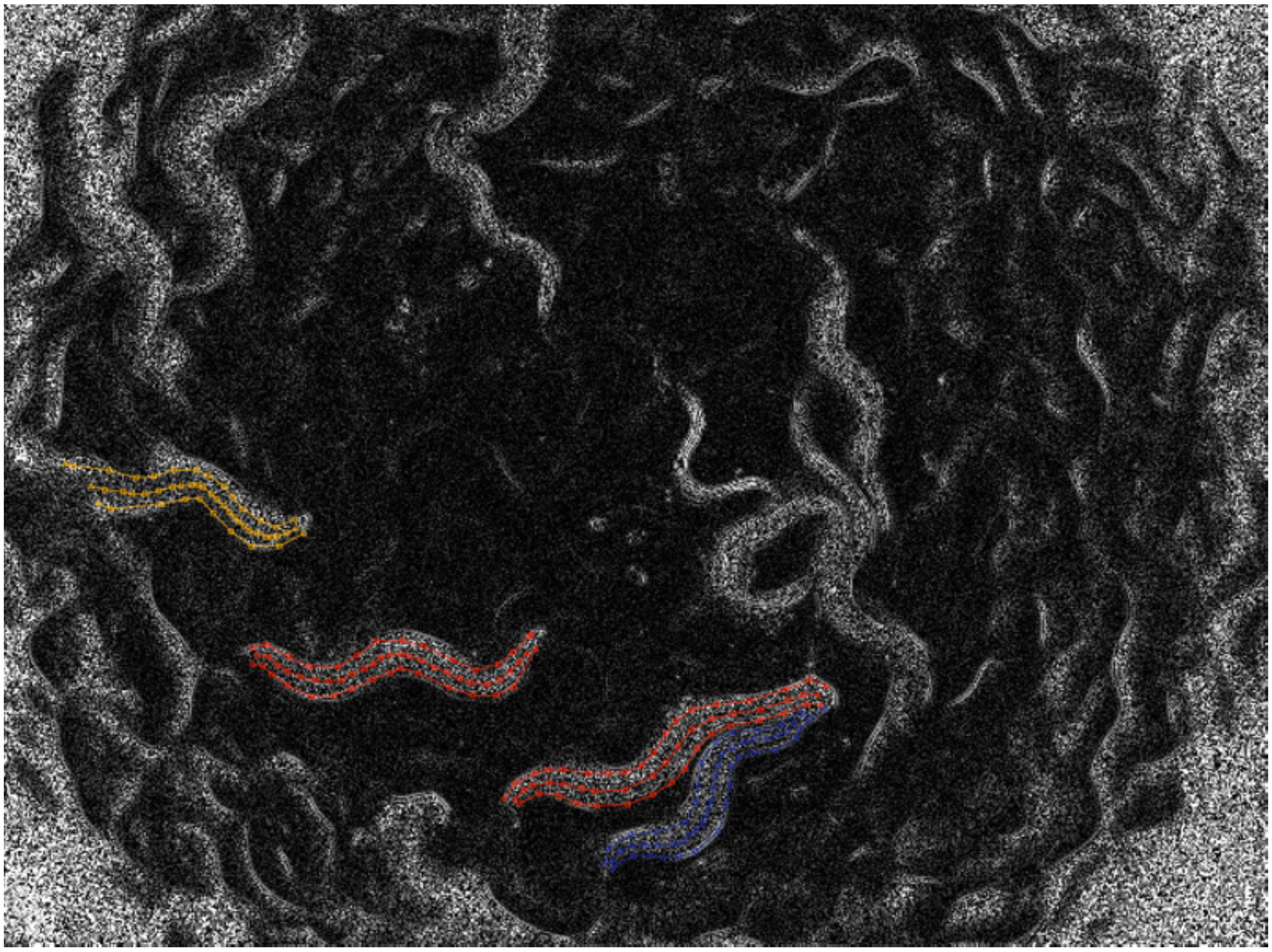
Preprocessing: Edges



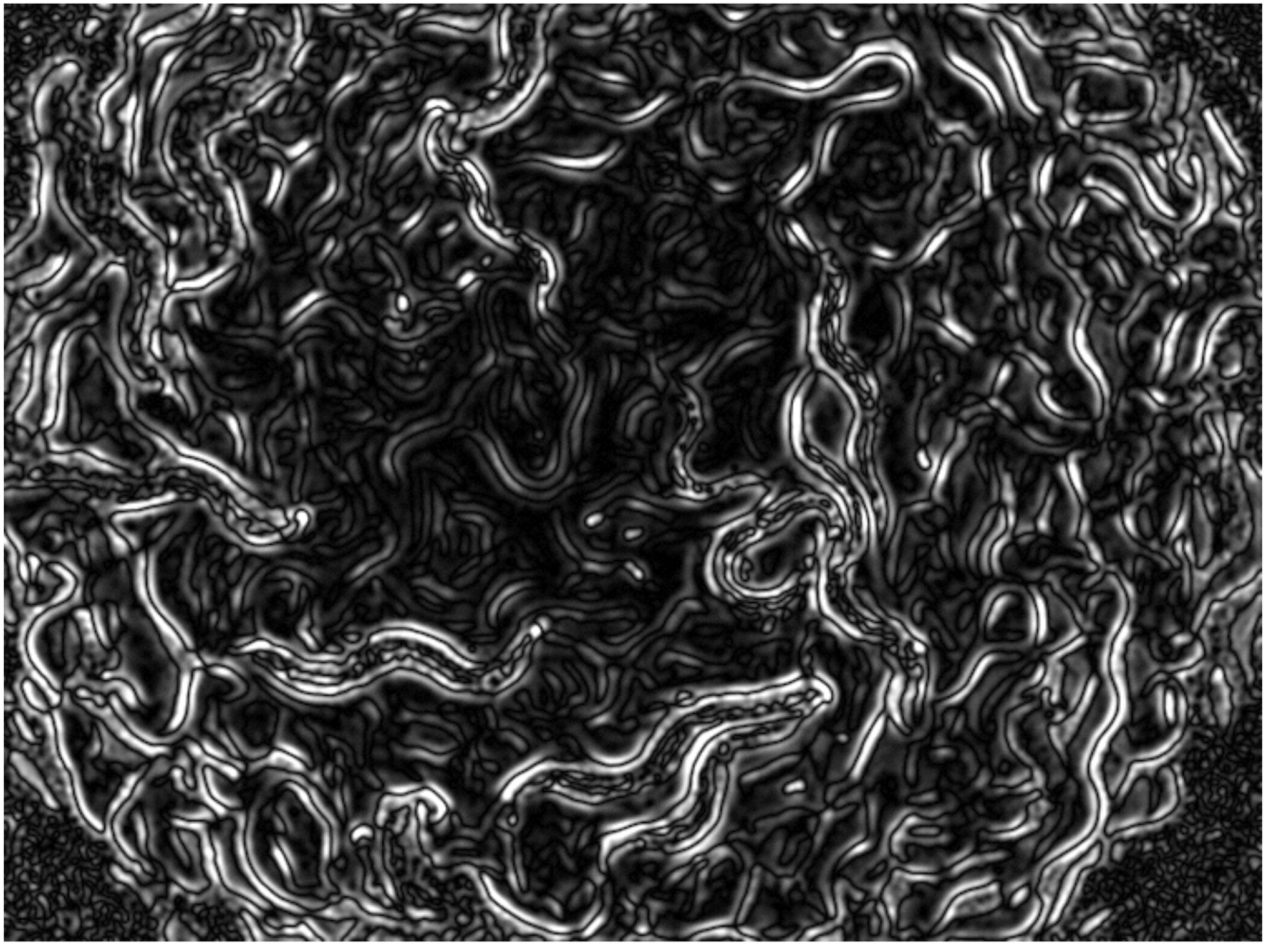
Preprocessing: Laplacian of Gaussian (I)



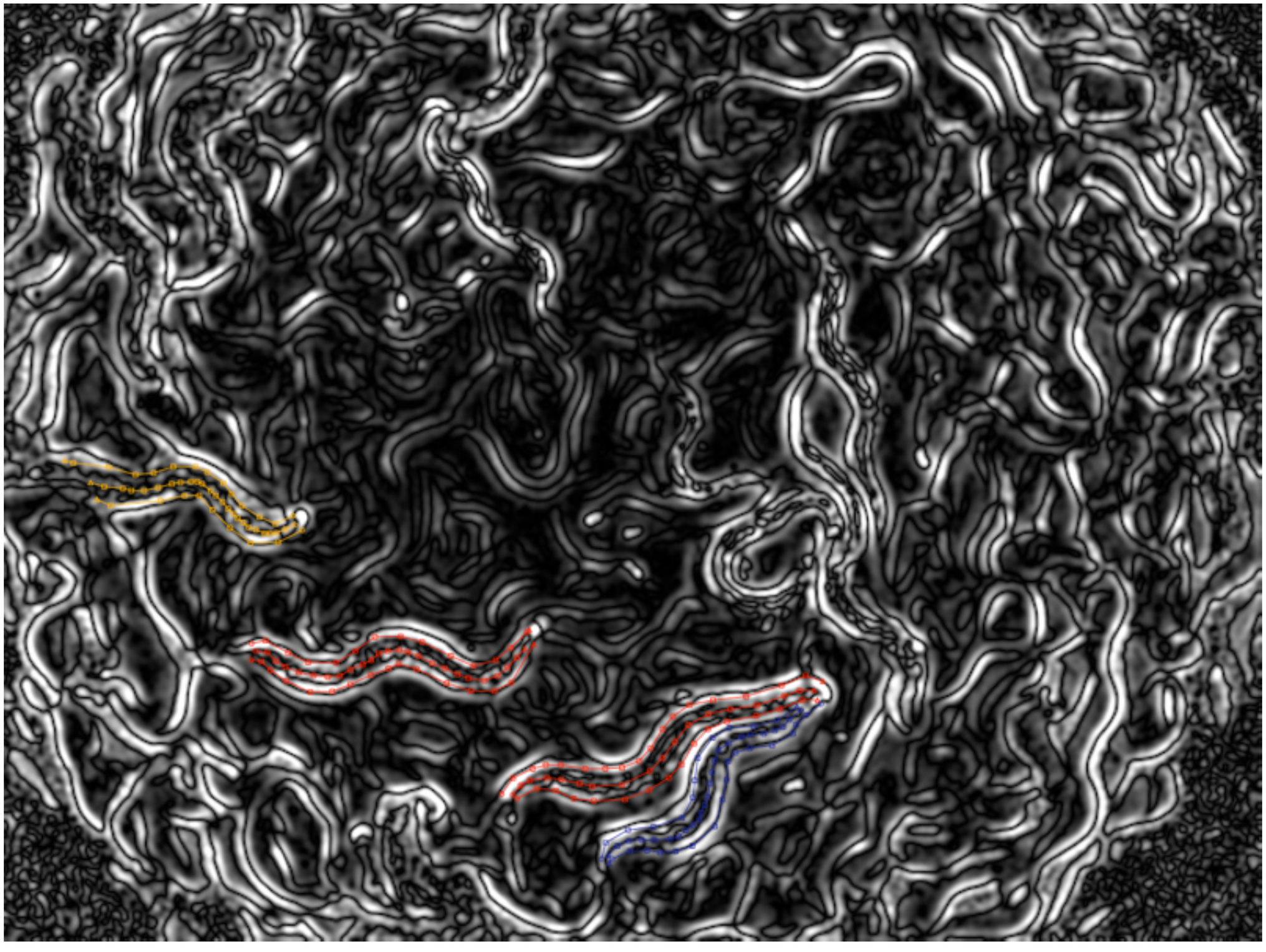
Preprocessing: Laplacian of Gaussian (I)



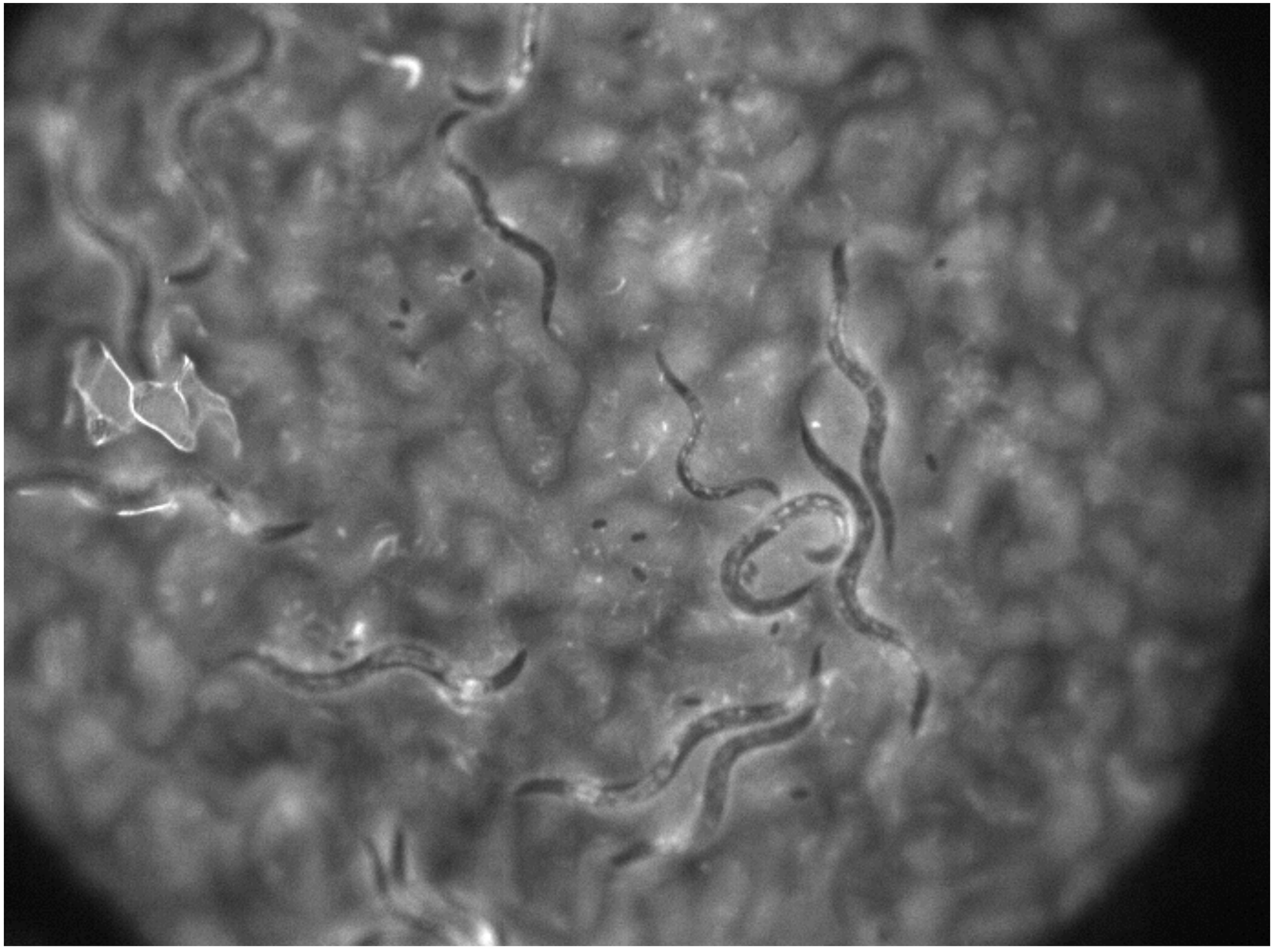
Preprocessing: Laplacian of Gaussian (II)



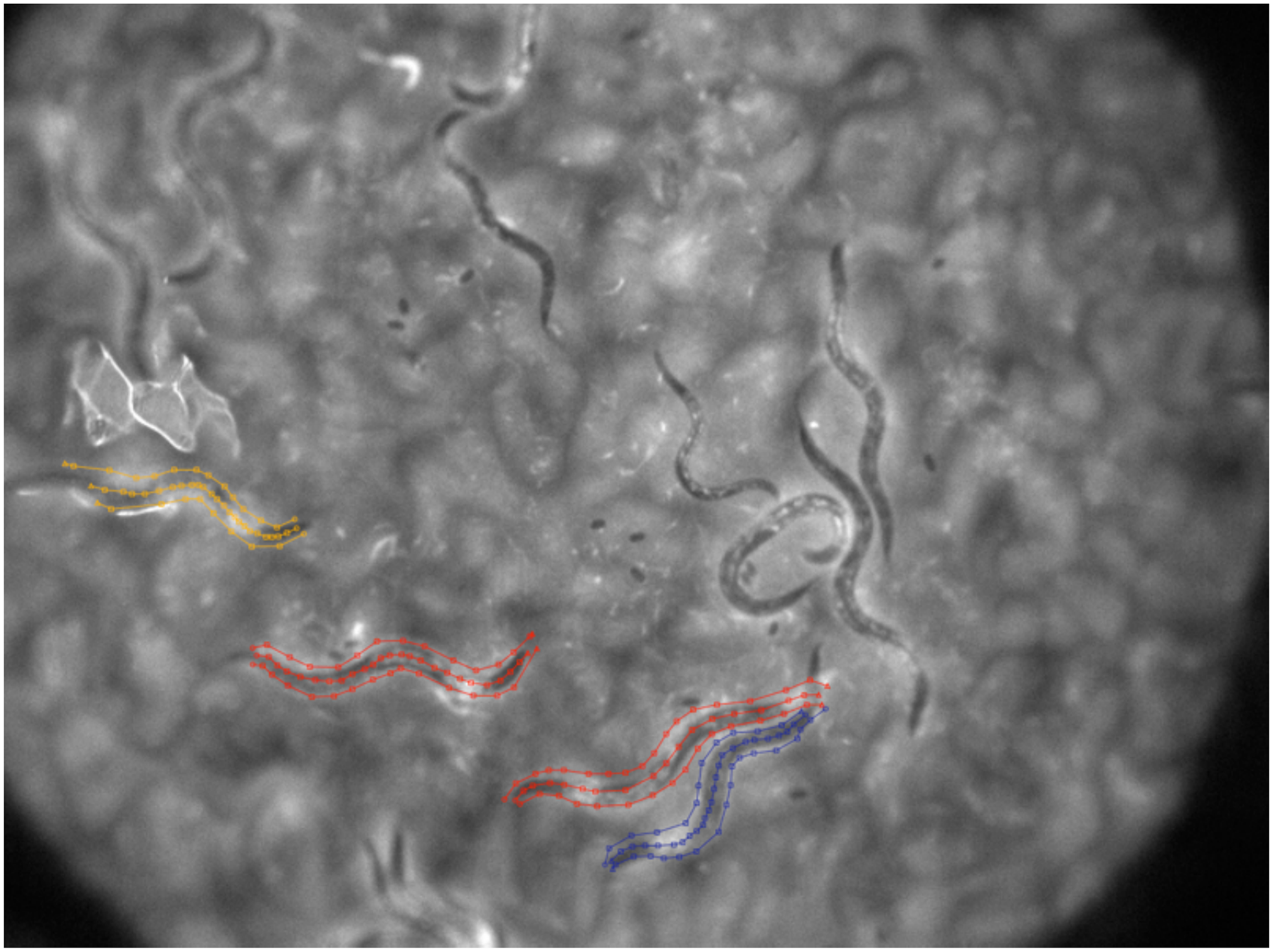
Preprocessing: Laplacian of Gaussian (II)

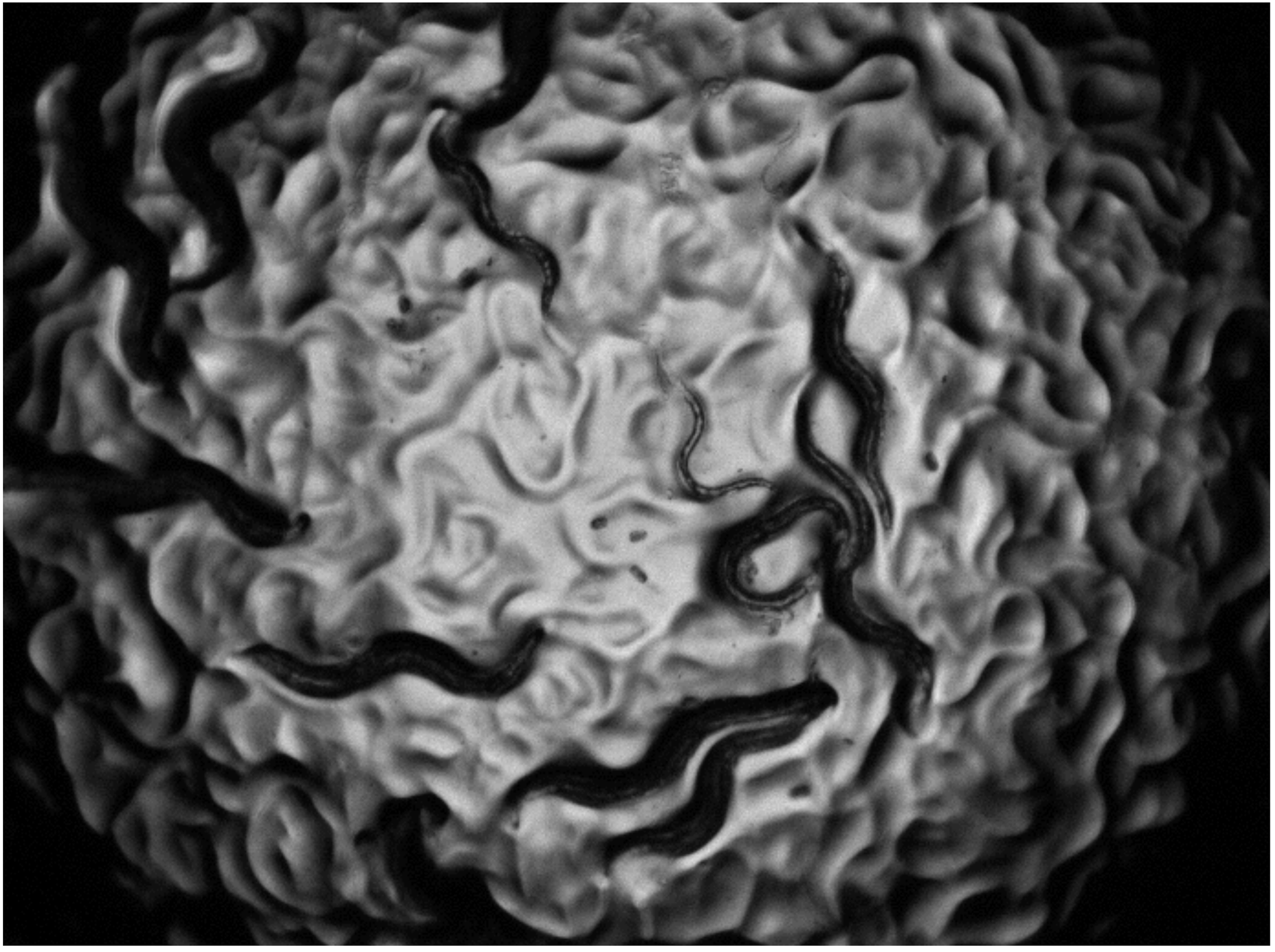


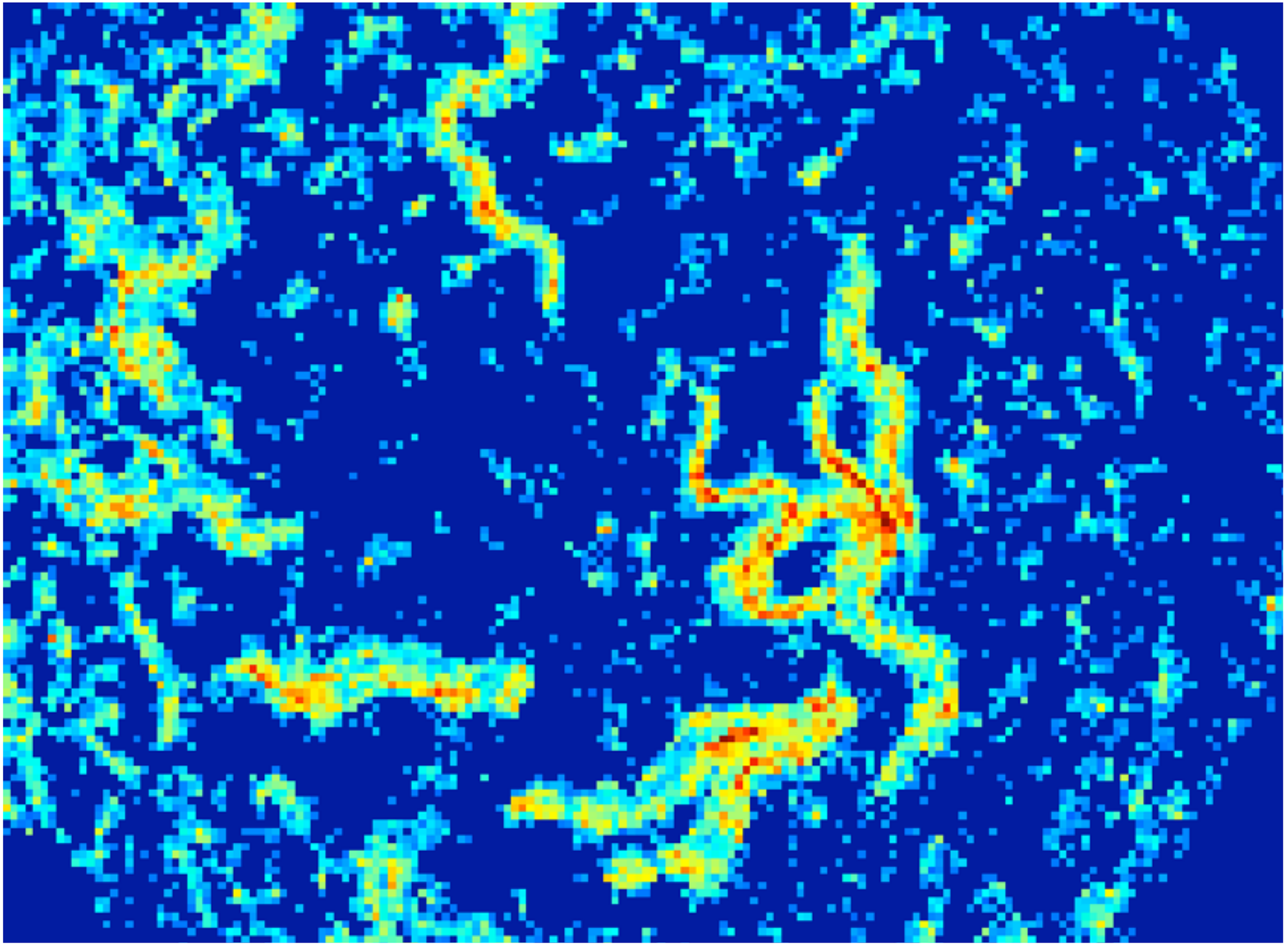
Fluorescent Image



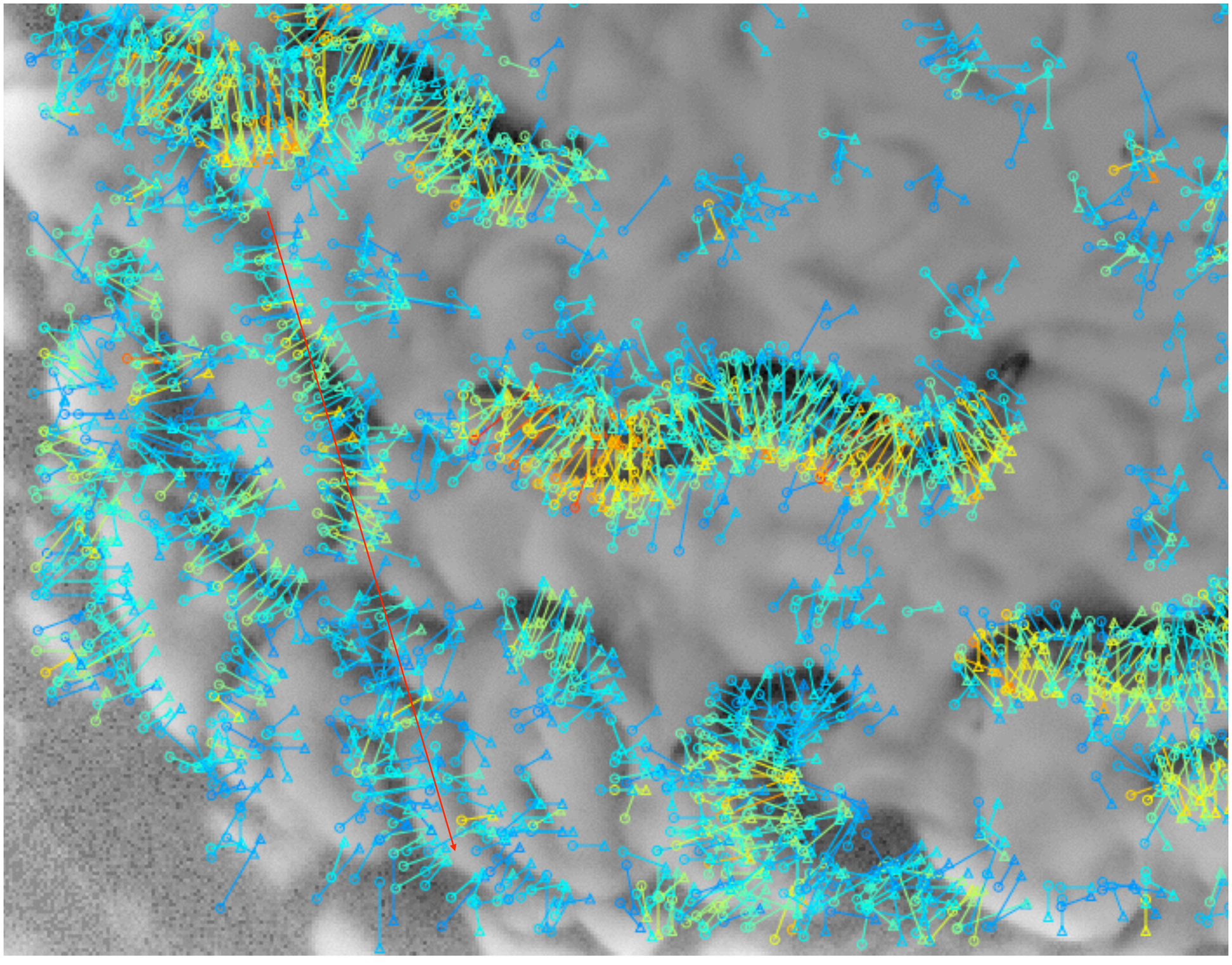
Fluorescent Image

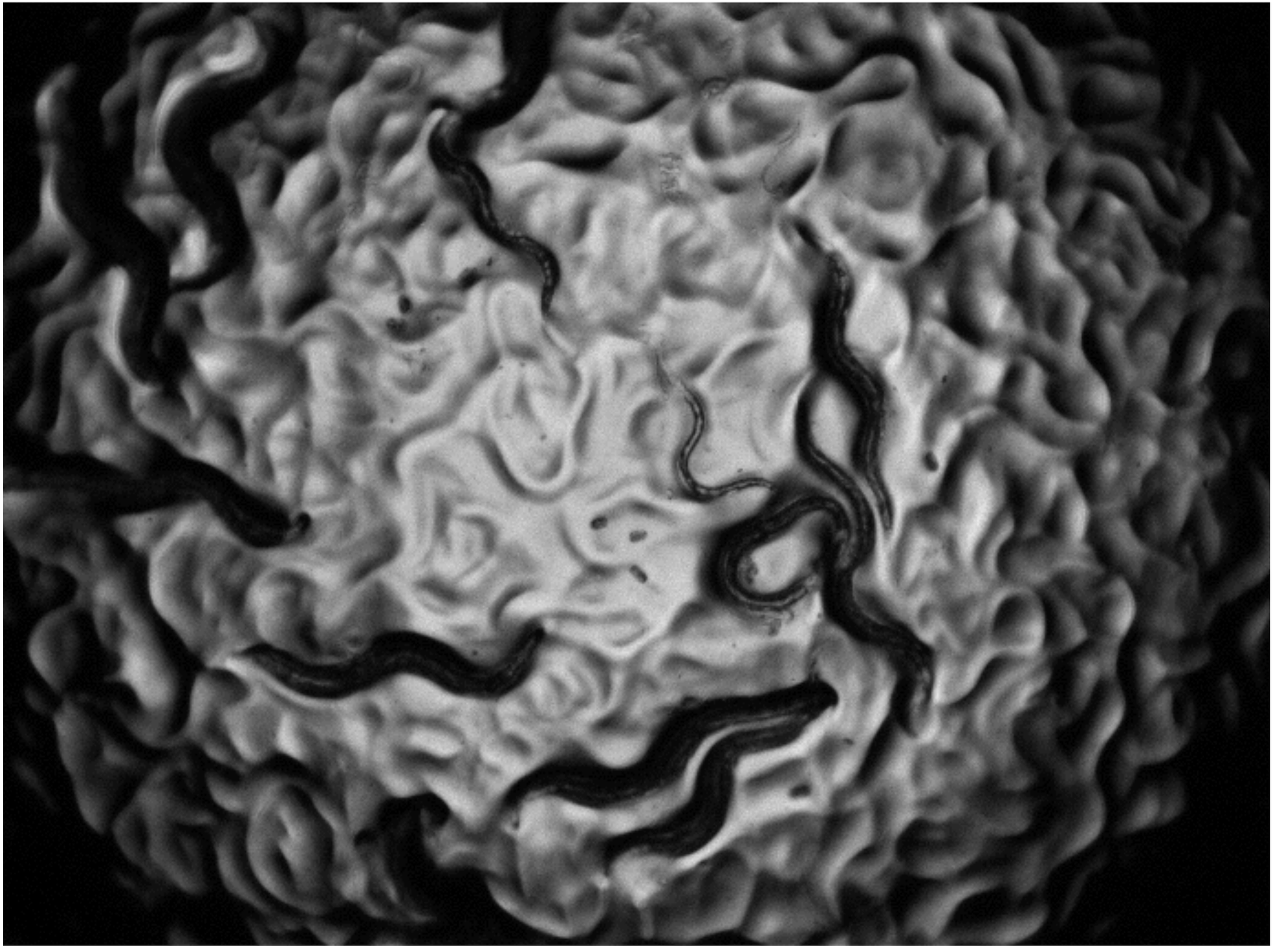


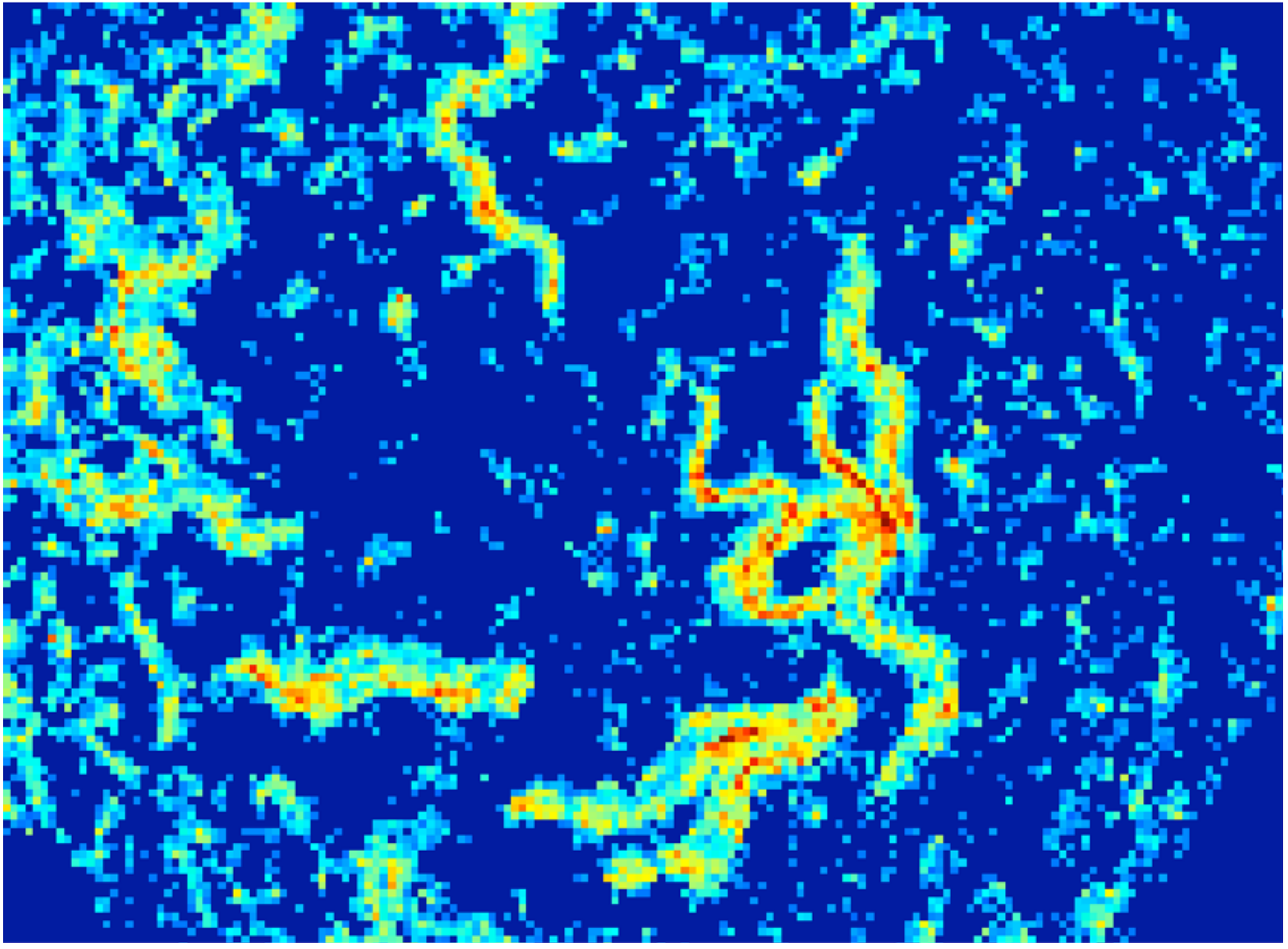


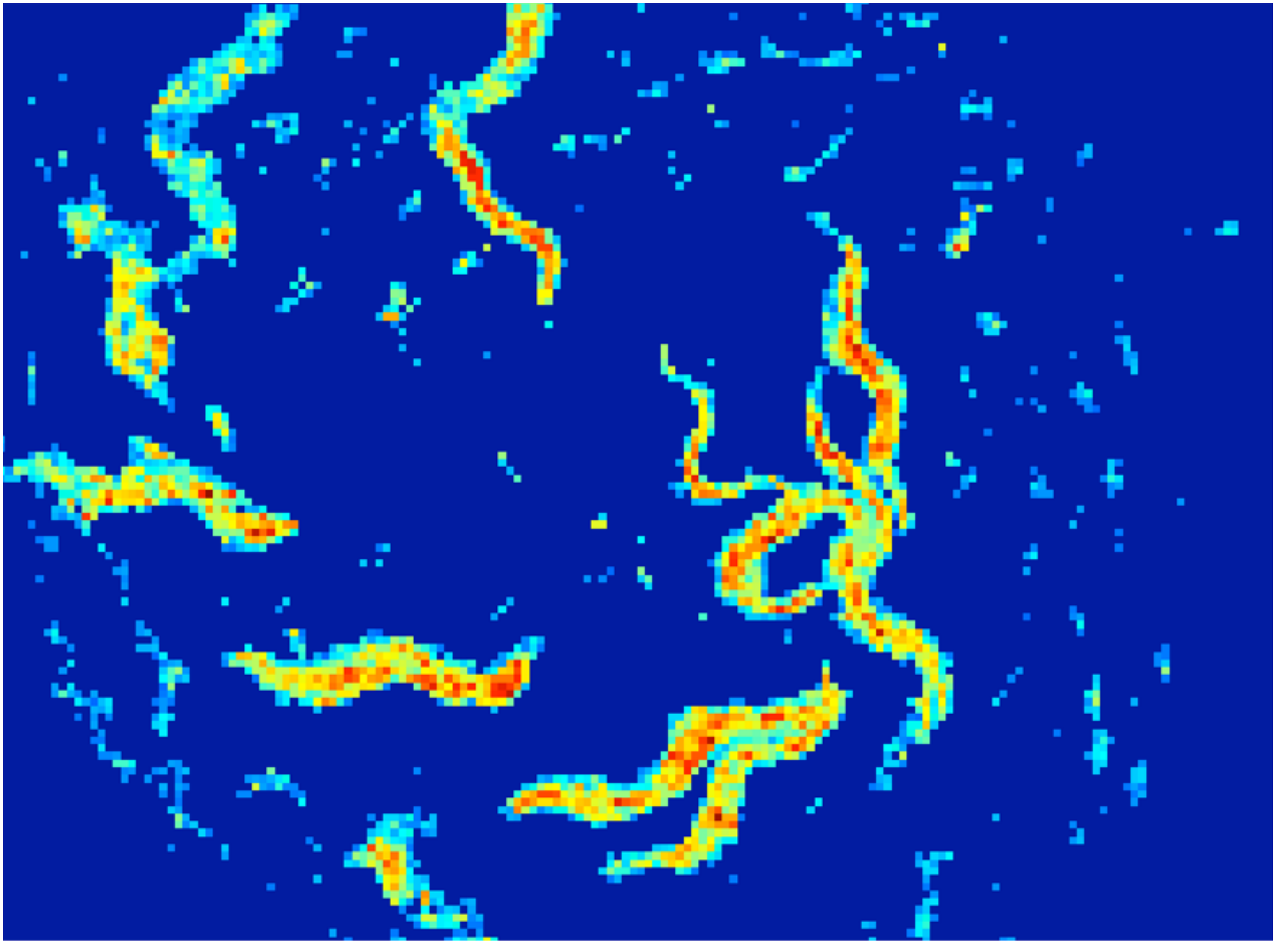


Relabeling false detections

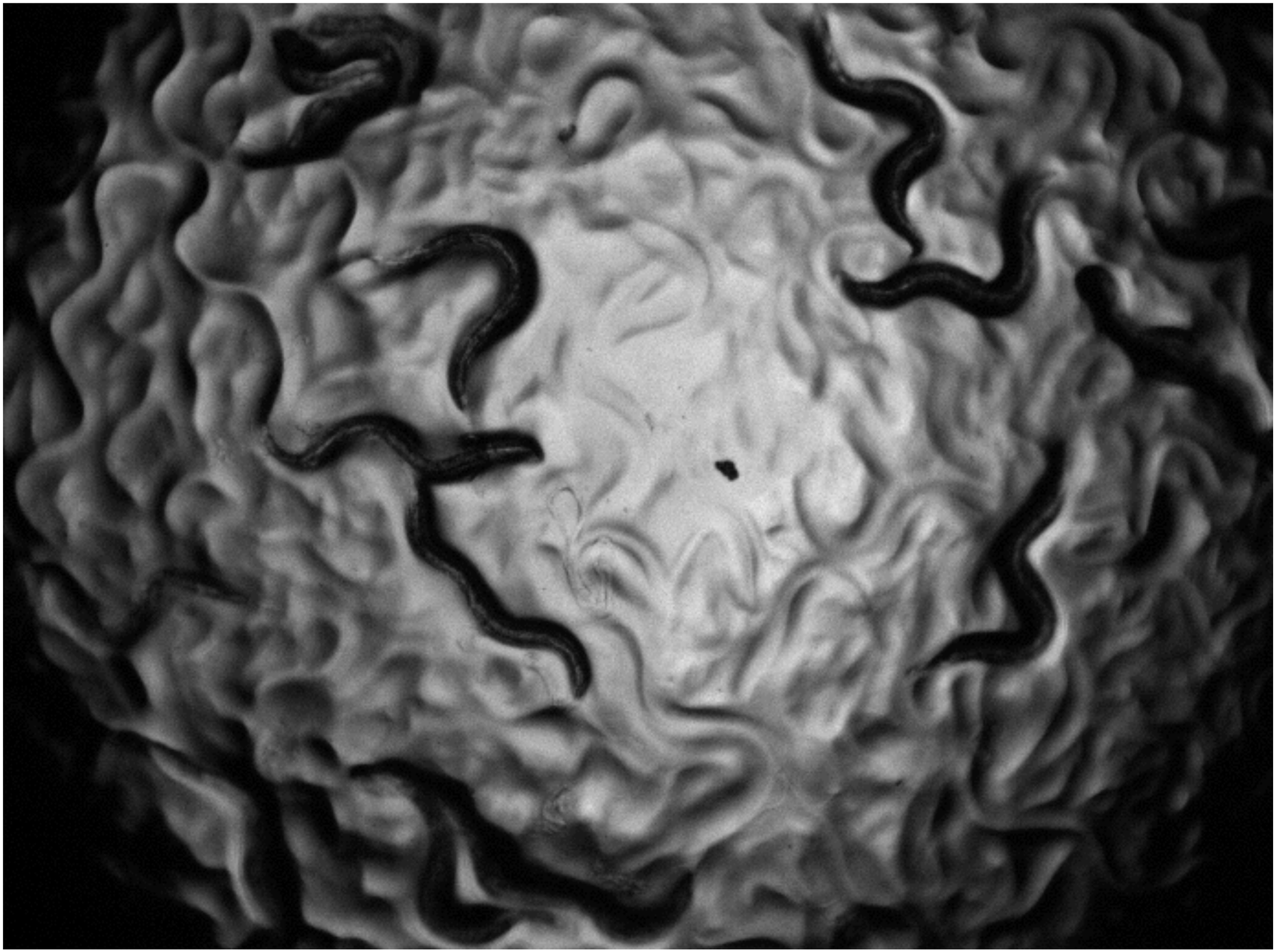




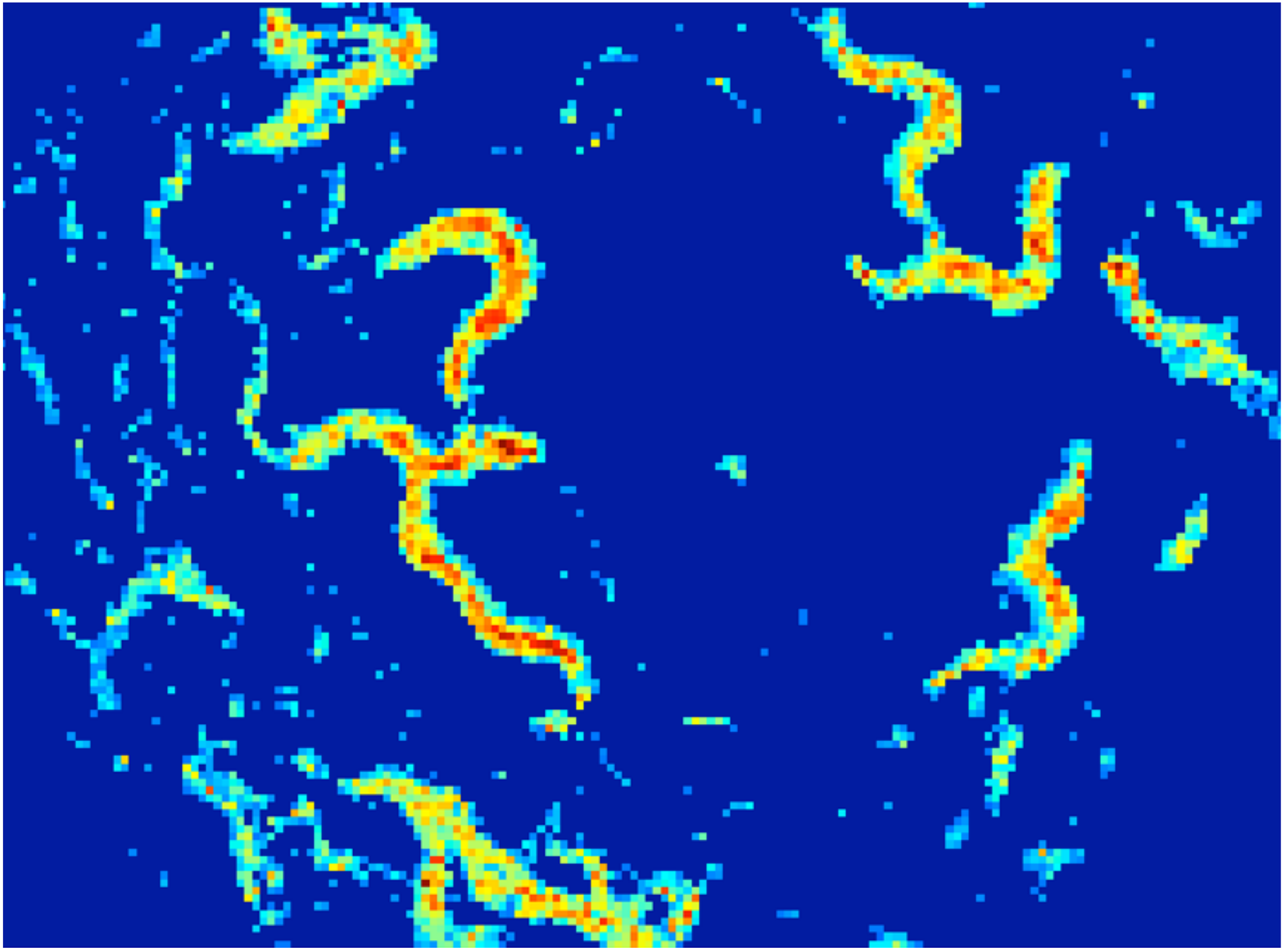




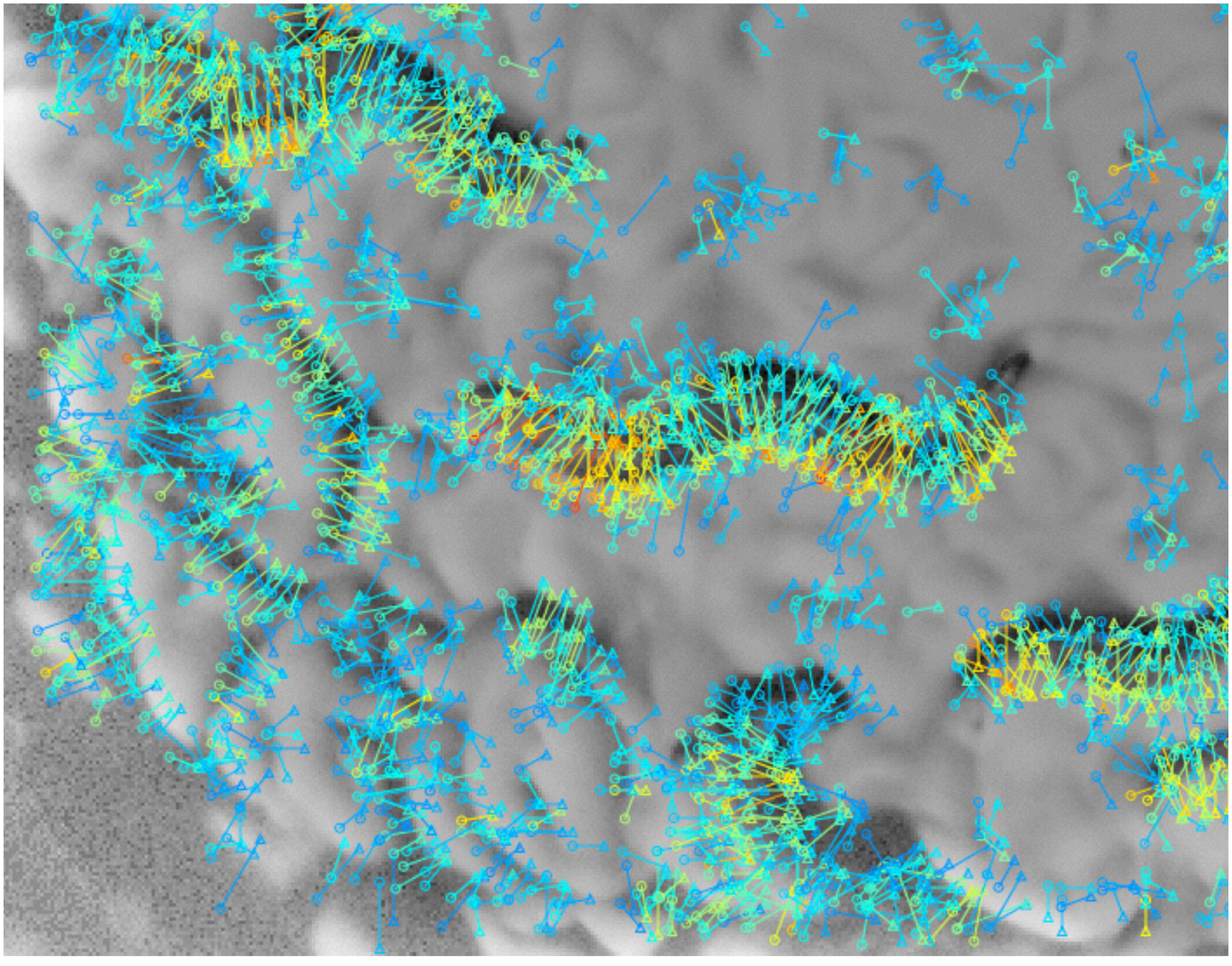
Results - Test Image



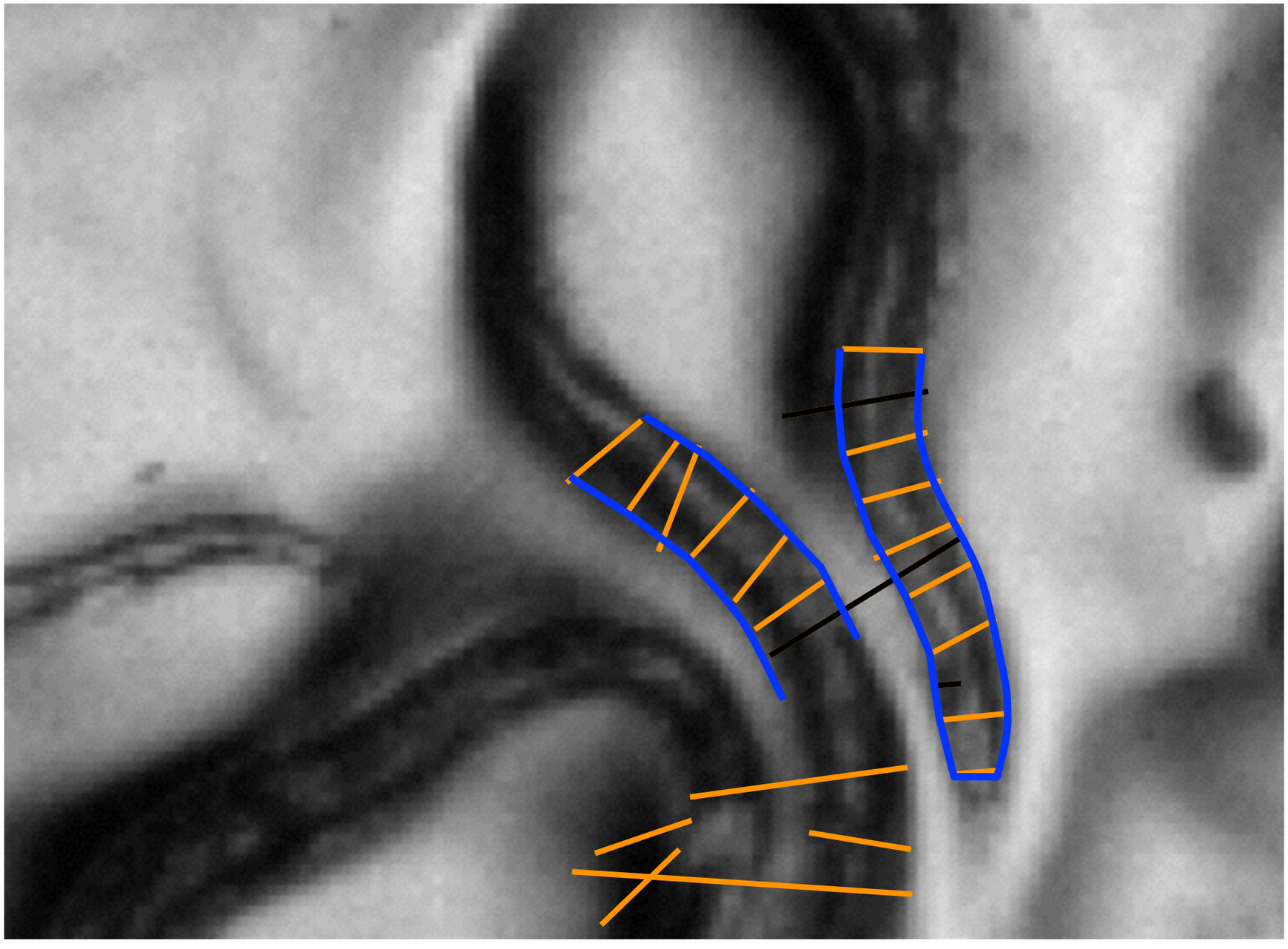
Results - Test Image



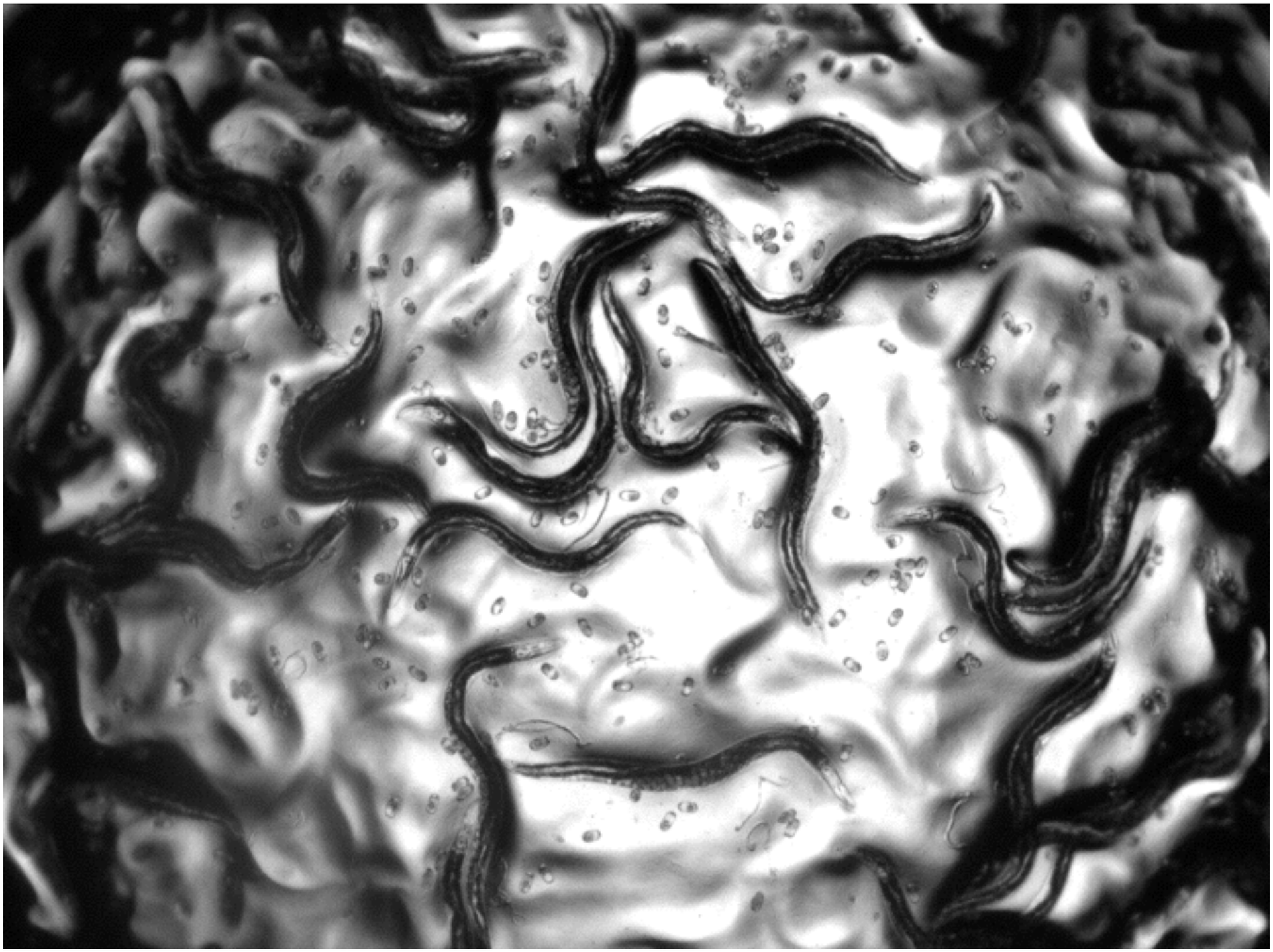
Relabeling false detections



The plan

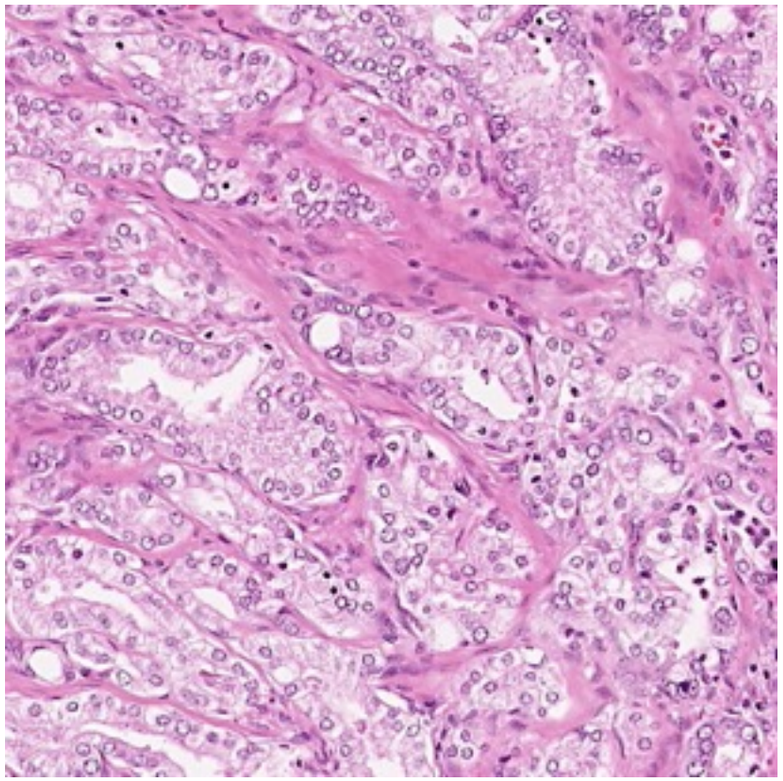


But wait... there’s more!

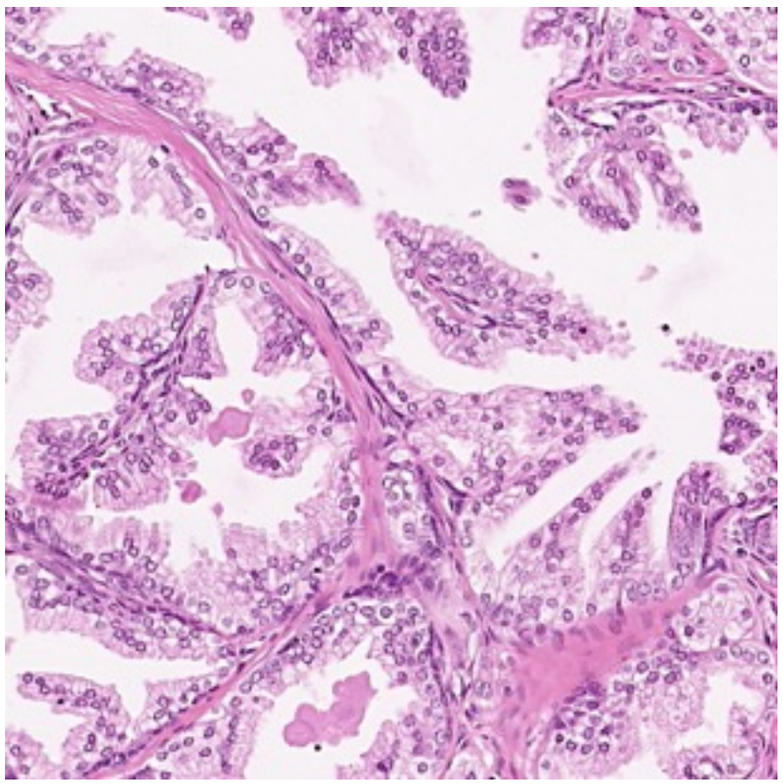


Future Work

- Differentiate between hyperplasia and cancer.
- Hyperplasia is very common in older men.
- It is benign and Pathologists can distinguish between hyperplasia and cancer at low magnification.
- If hyperplasia is marked as cancer then not much savings.

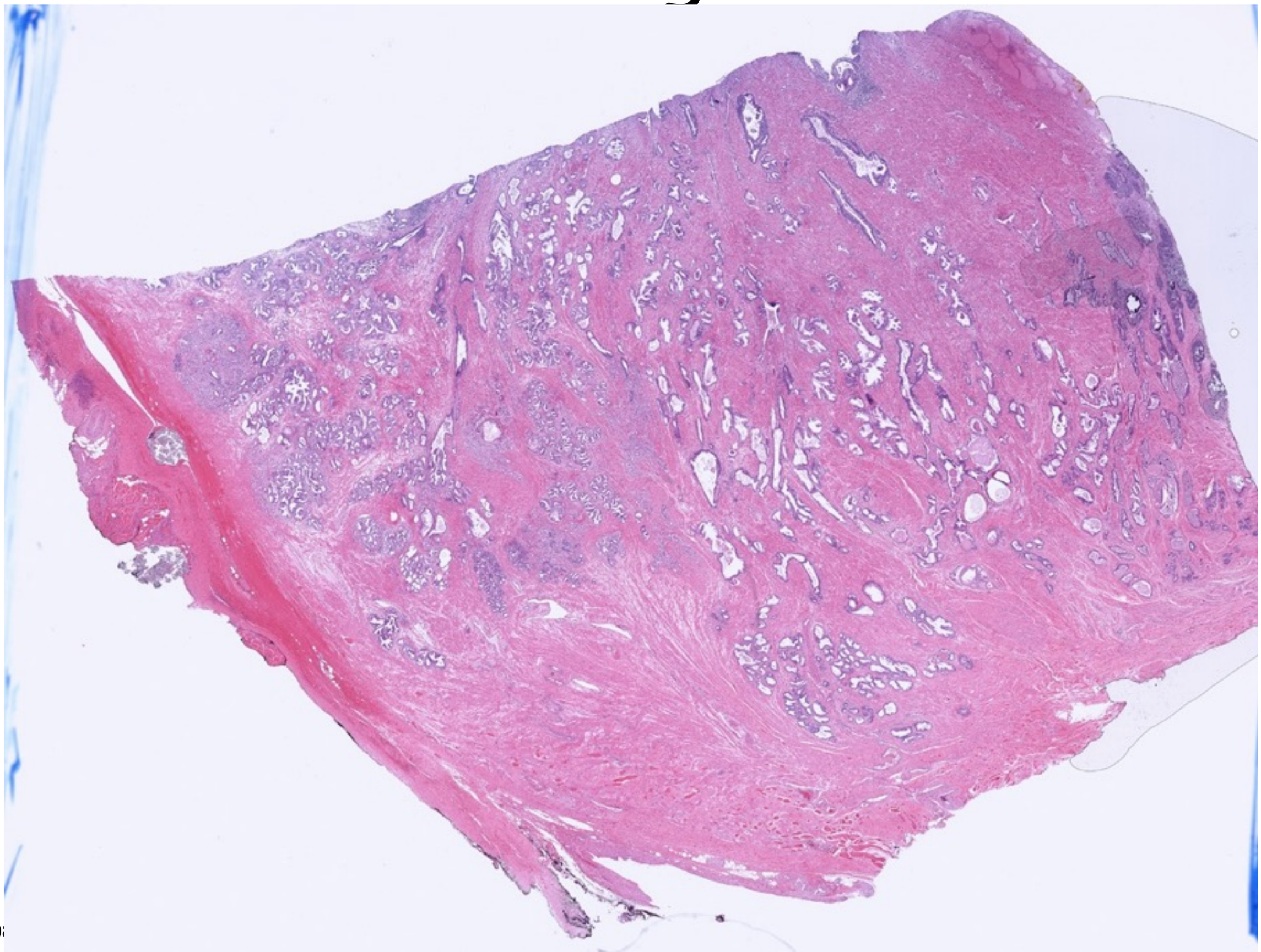


Cancer



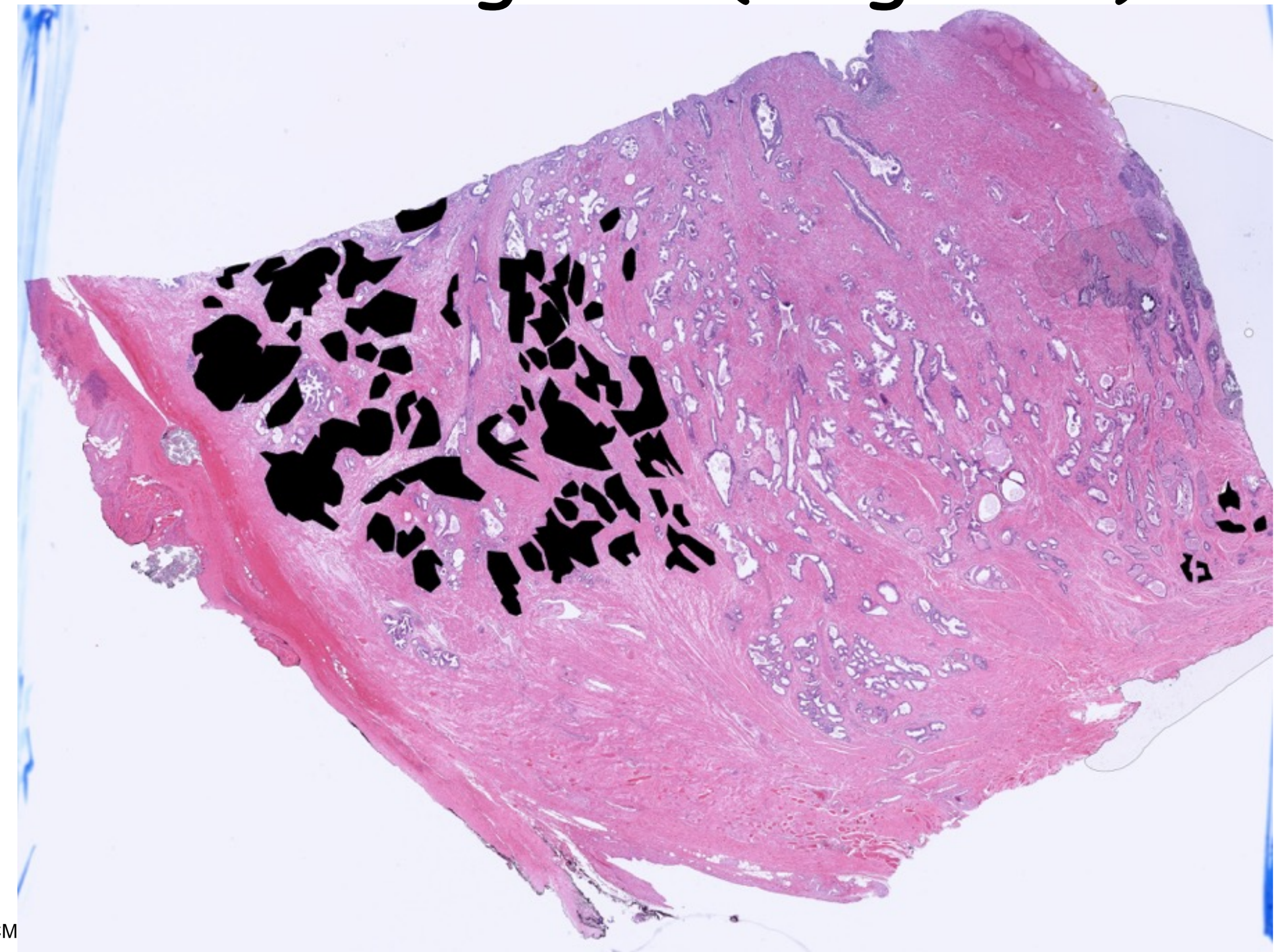
Hyper-plasia

Training Set



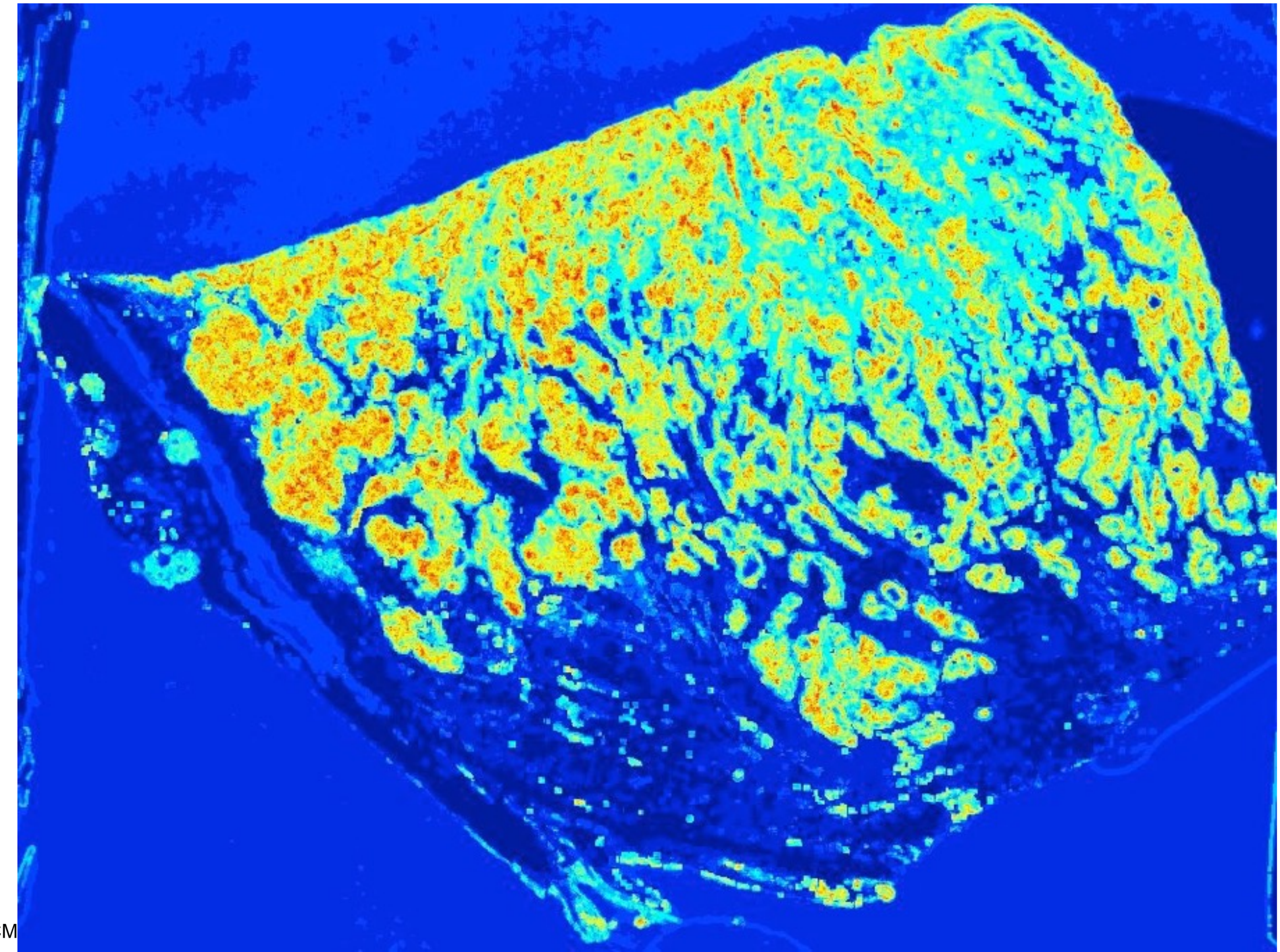
Training Set (Positive)

Training Set (Negative)



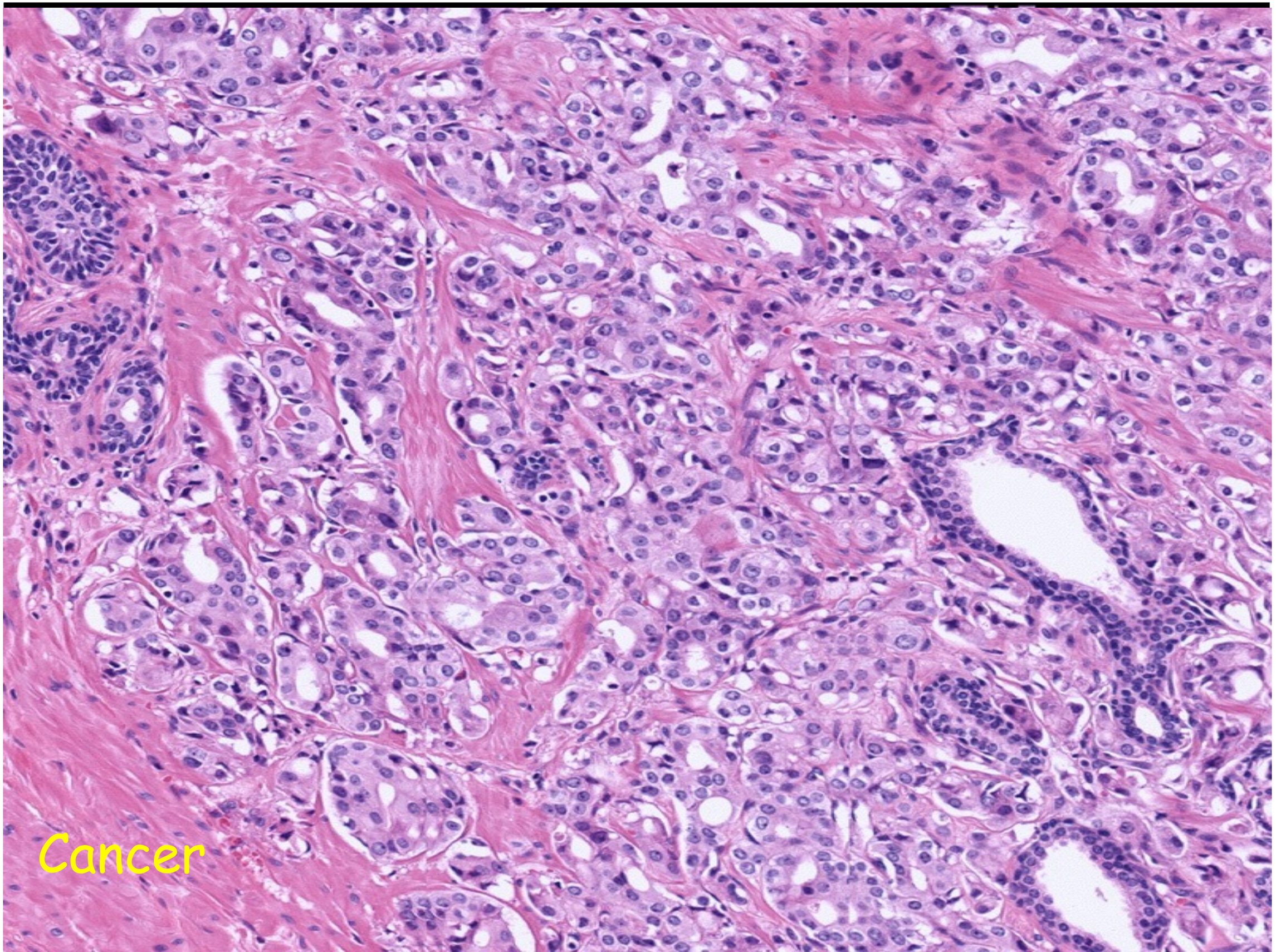
ECM

Results

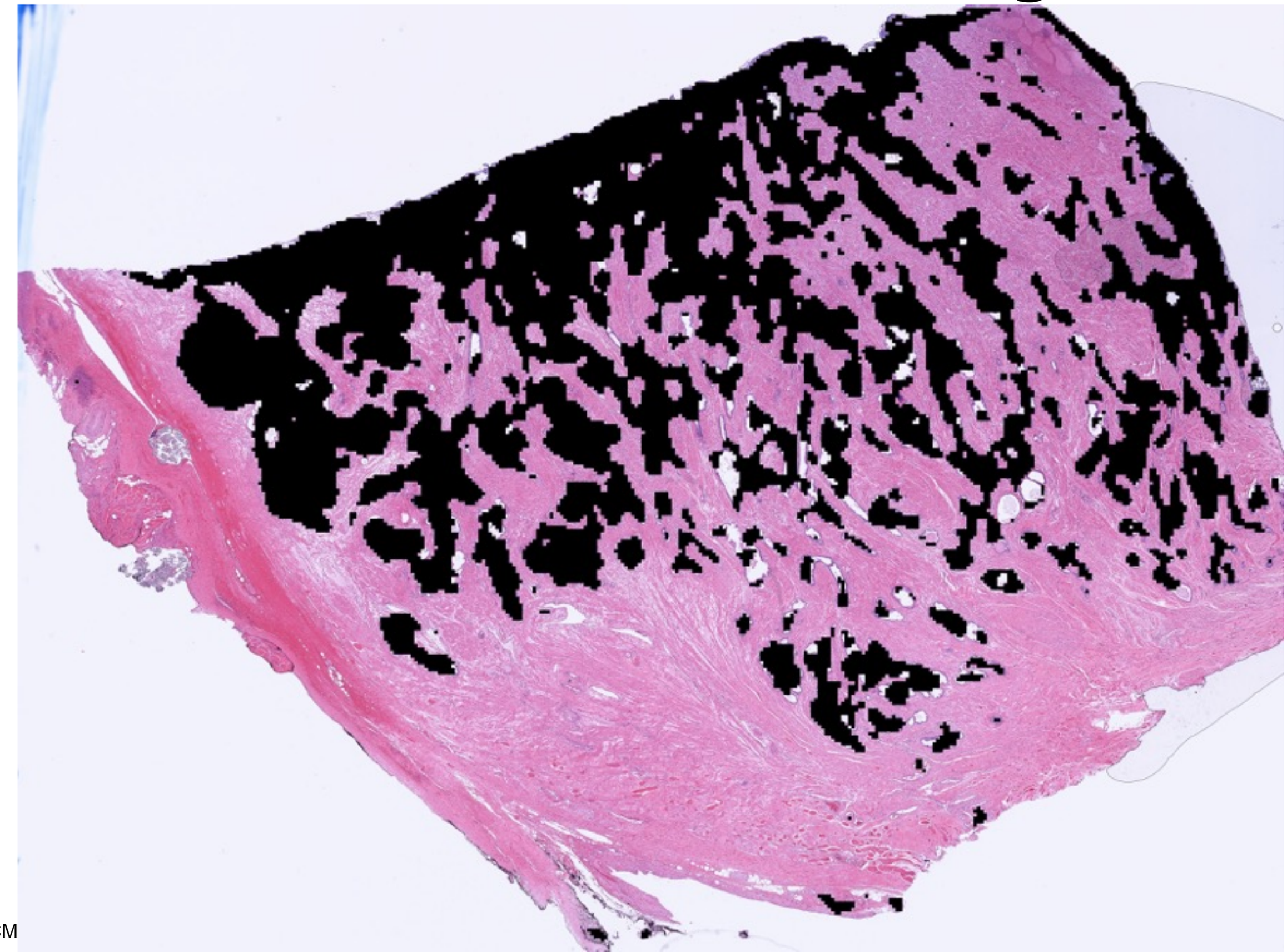


ECM

Results - Predicted Positives

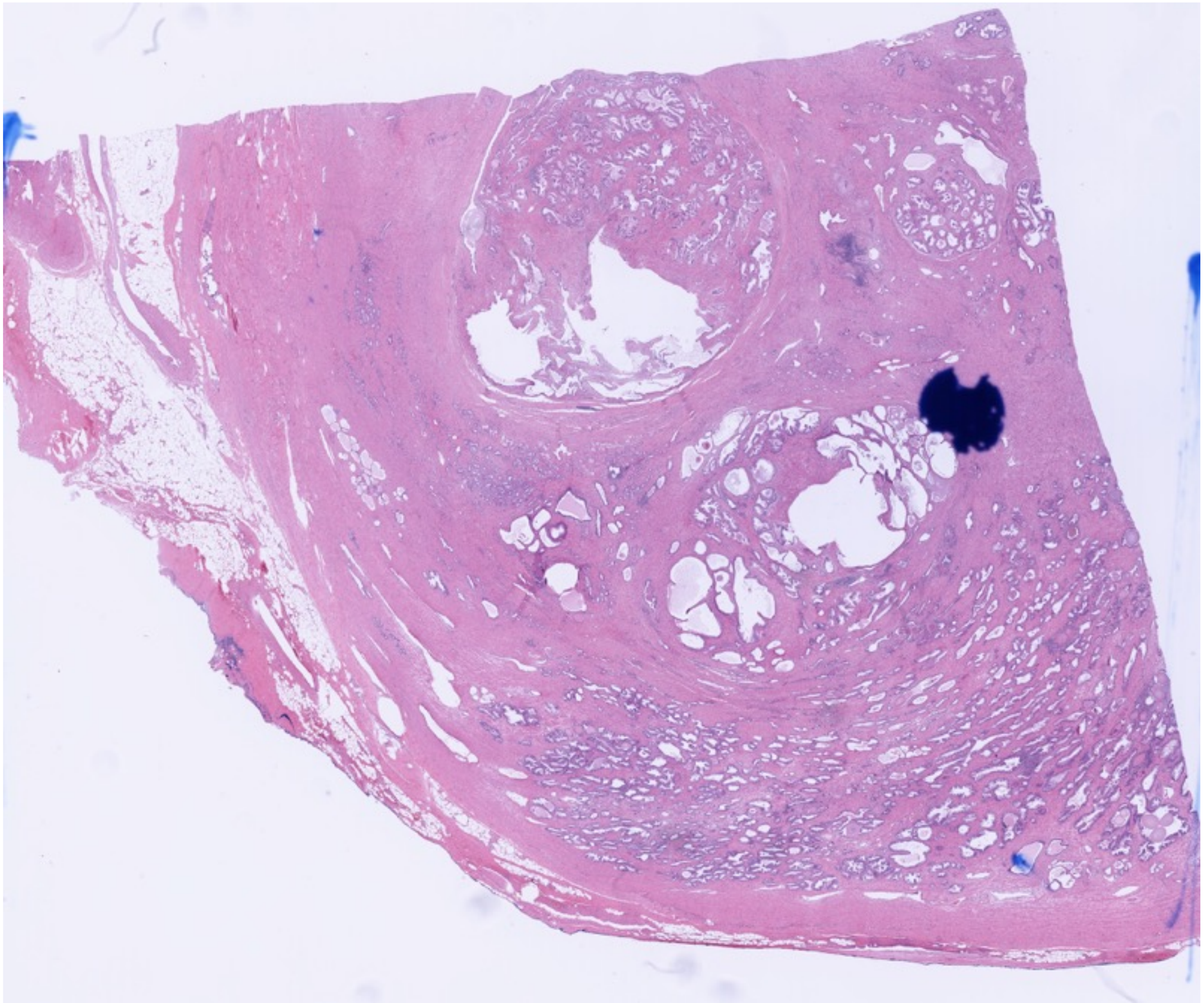


Results – Predicted Negative



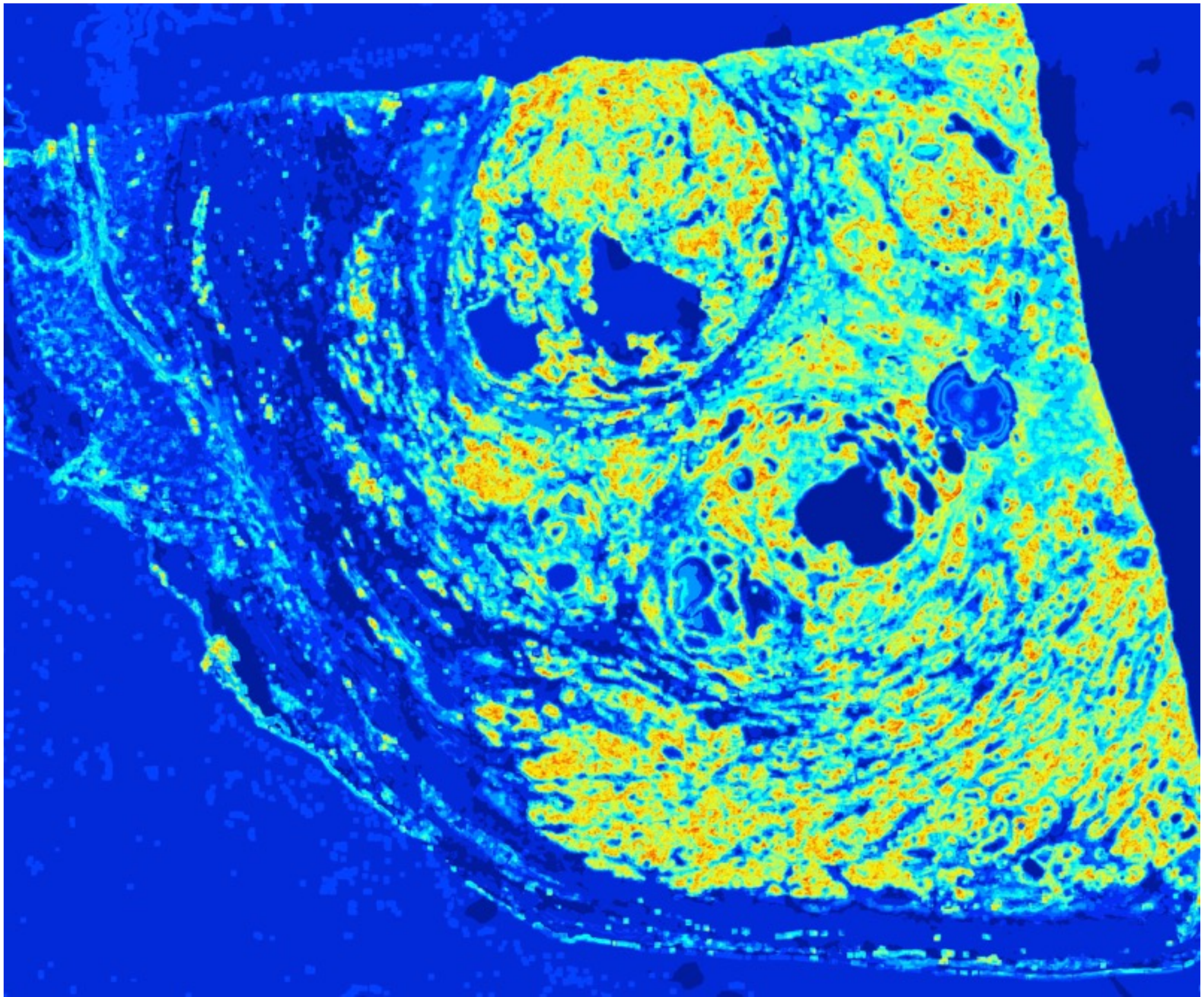
ECM

Results - Test Image

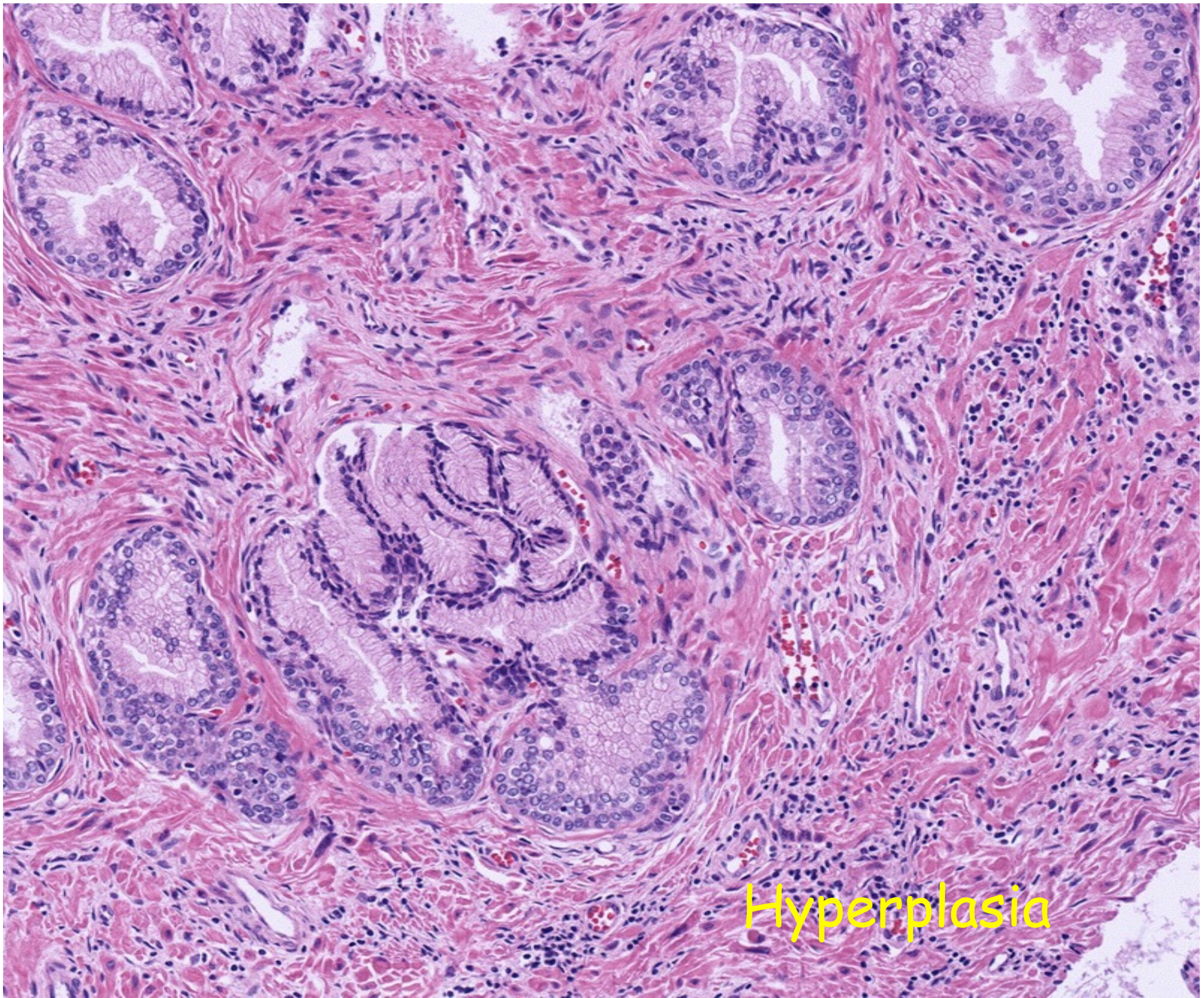


ECML08

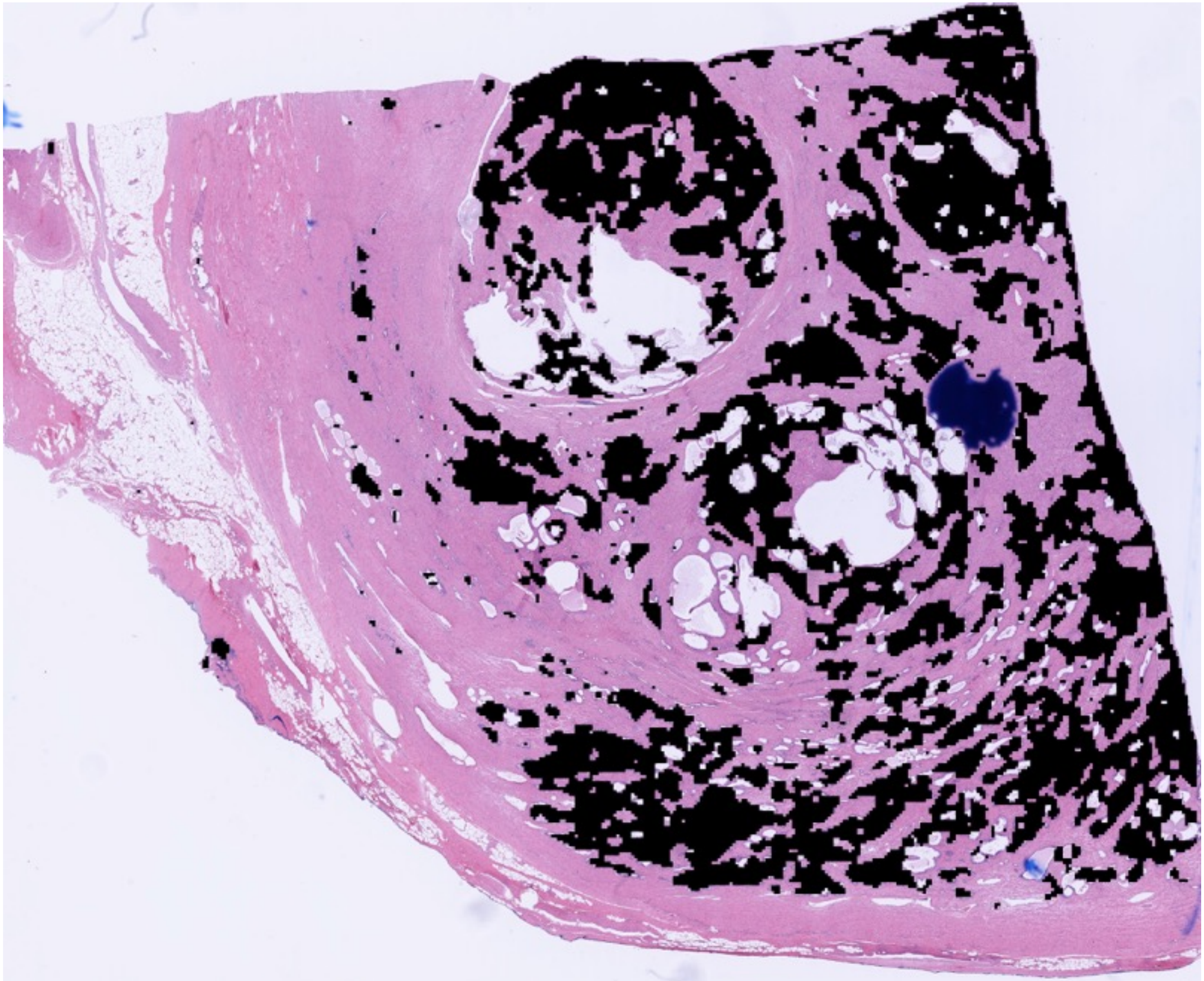
Results Test Image - Scores



Results Test Image - Positives

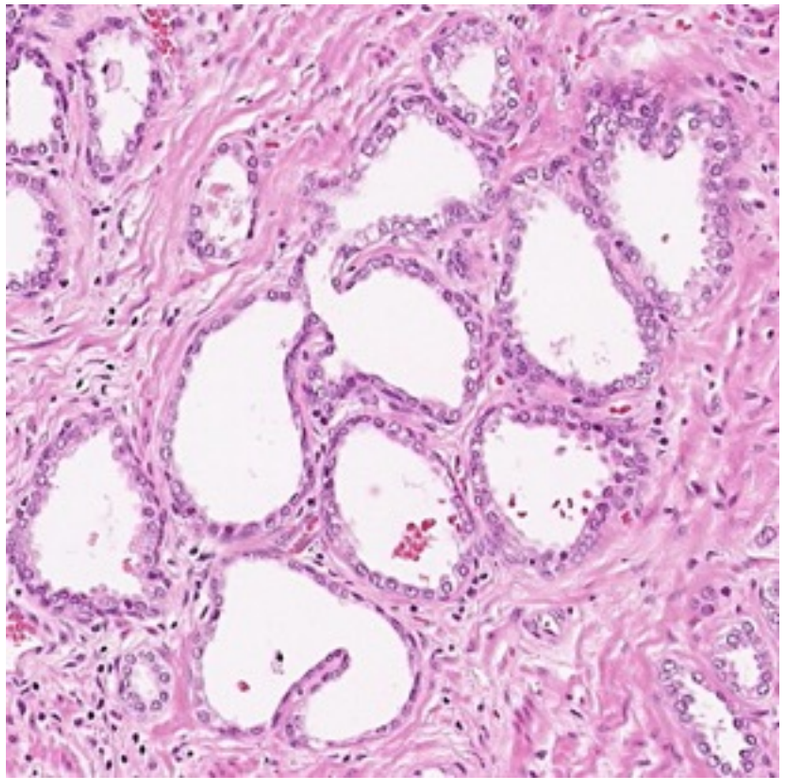


Results Test Image - Negative

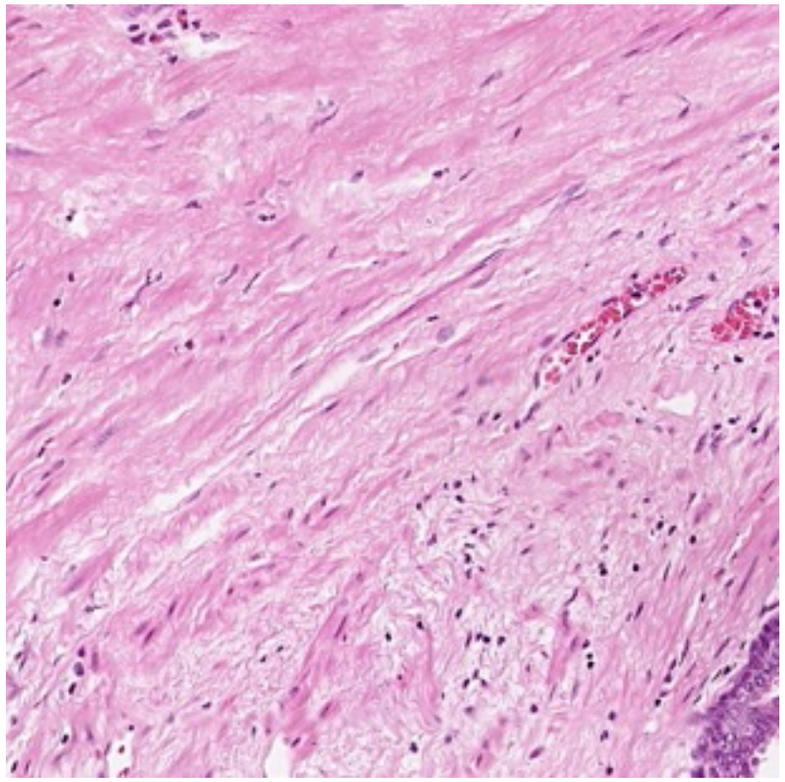


ECML08

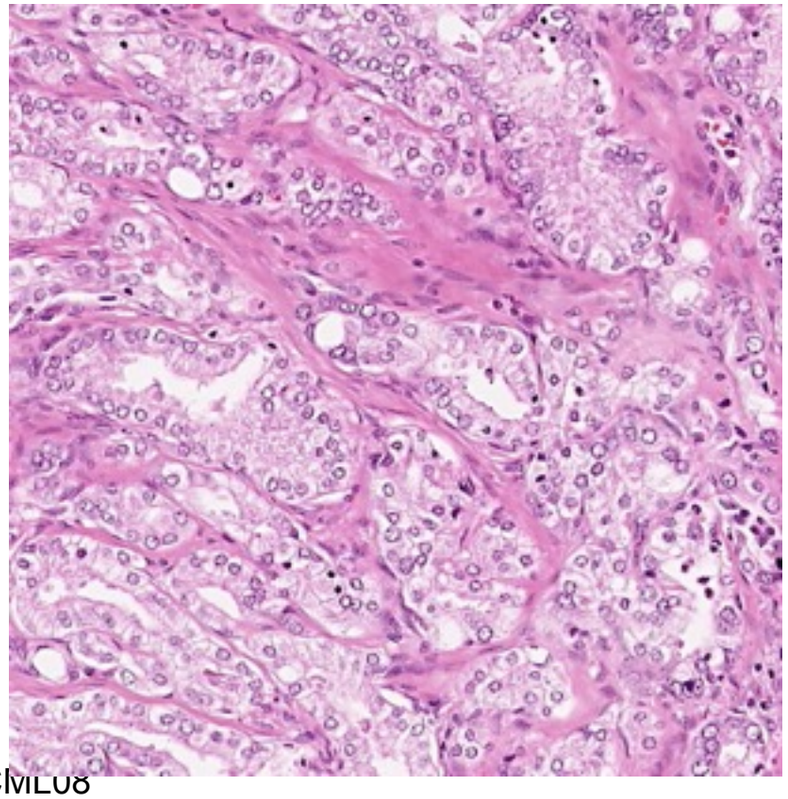
Examples



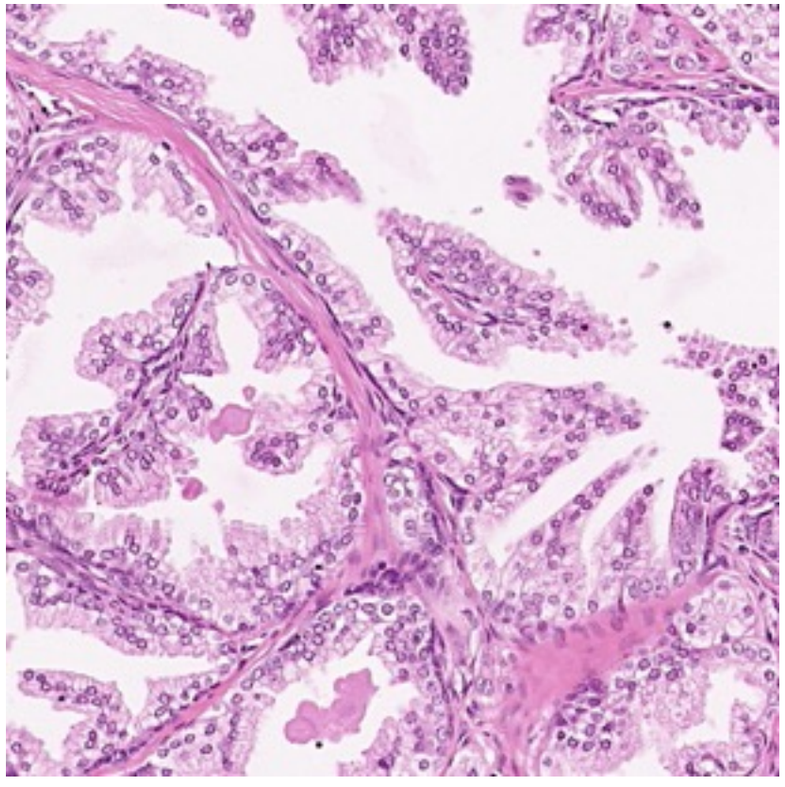
Healthy Glands



Connective Tissue



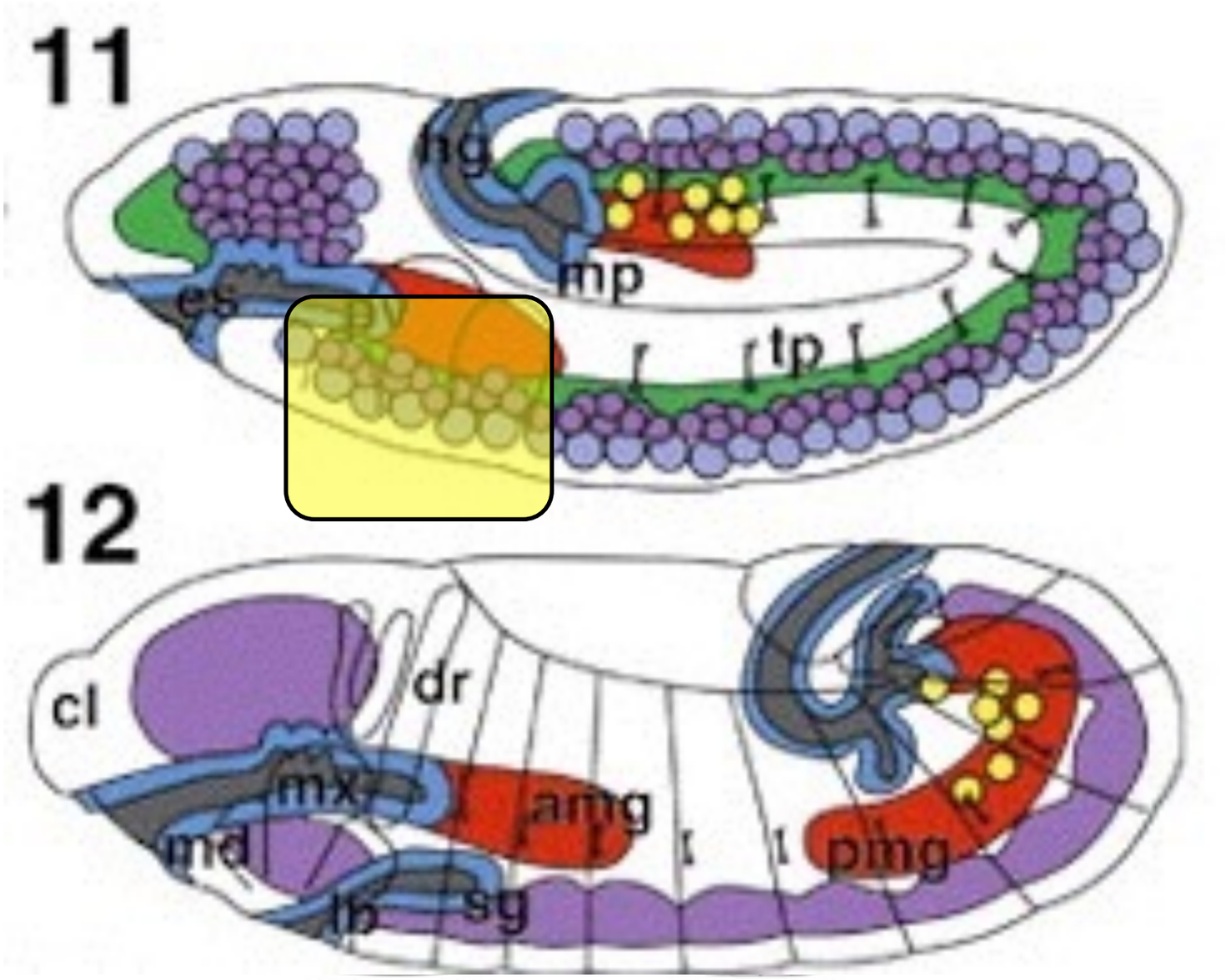
Cancer



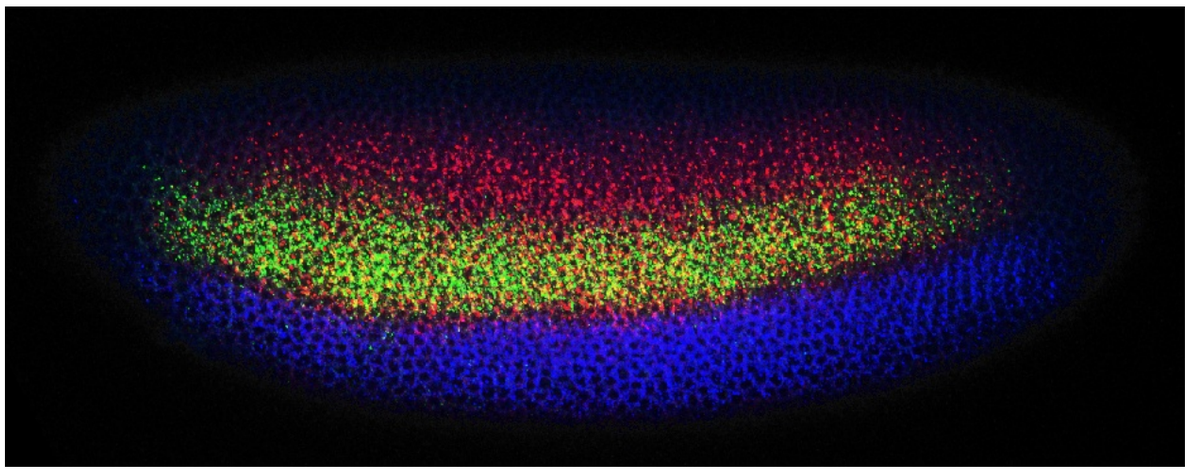
Hyperplasia

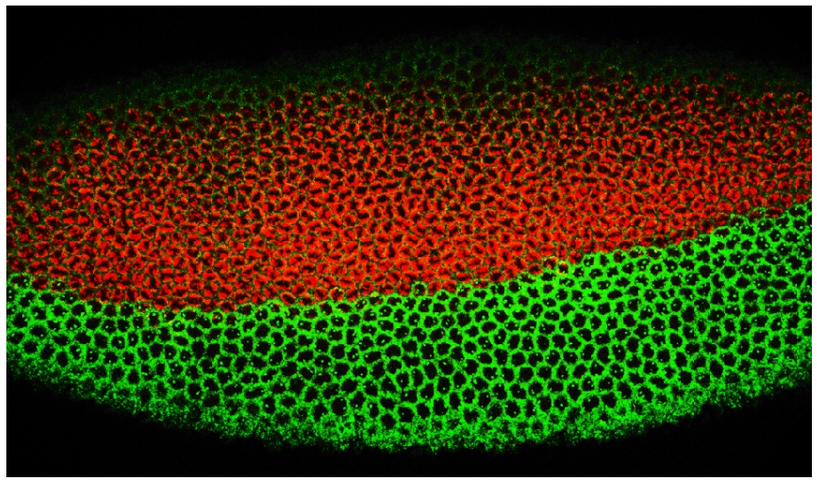
Regulation of development in Drosophila Embryos

McGinnis Lab, UCSD, Gary Tedeschi



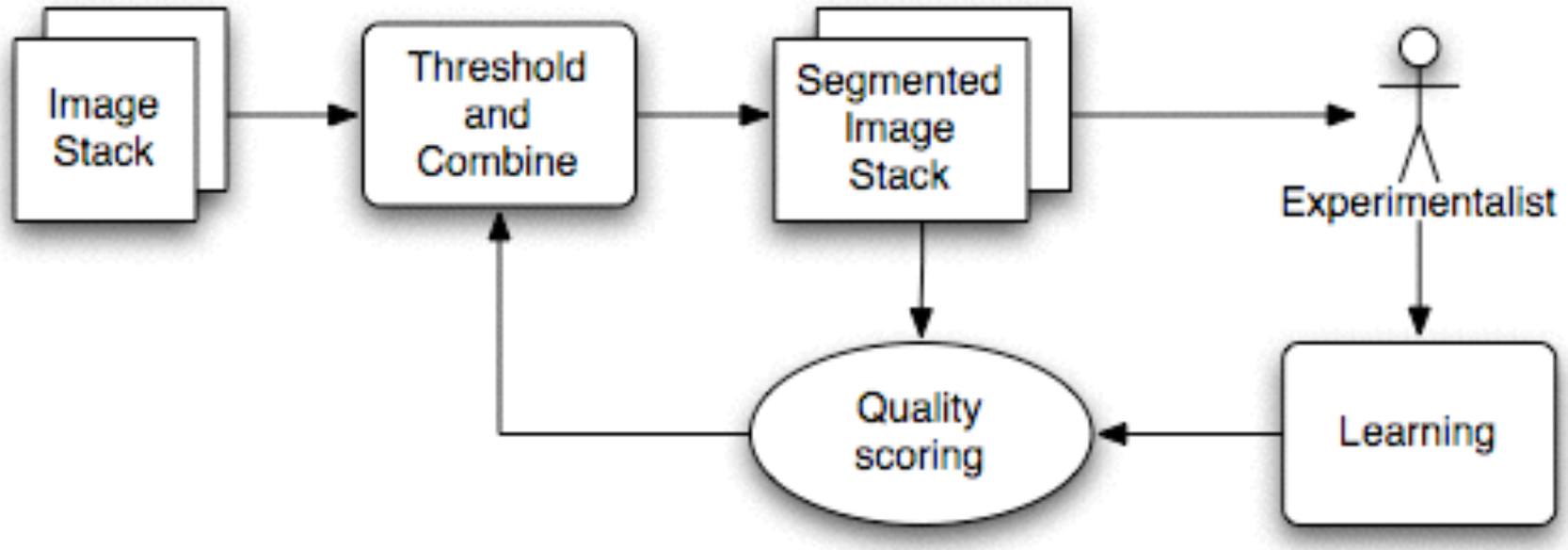
SNA boundary is amazingly sharp and straight





- SNA is one of the earliest structuring genes
- It is regulated by concentration level of a regulator gene (forgot name)
- Concentration of regulator is very low.
- Simple regulation would lead to “salt and pepper” around boundary?
- How do we get such straight boundaries?
- Must involve some inter-cell feedback mechanisms

Basic workflow



Manual feedback