Computer algebra: Formalization and applications to network reliability and biomedical image processing

Guanajuato, Nov 20th 2018

Eduardo Sáenz de Cabezón

Universidad de La Rioja (Spain)

UNIVERSIDAD DE LA RIOJA

Main research lines

1. New concepts and relations in the homological approach to commutative algebra and algebraic topology.

2. Implement homological algorithms, develop formal verification of algorithms, integrate both in usable systems.

3. Application of concepts and algorithms of homological algebra and algebraic topology to biomedical images analysis and system reliability

Goals

1. **Commutative algebra:** relations between ideals and simplicial complexes via polarization and depolarization, extend to persistence homology for ideals. Algorithms.

2. **Algebraic Topology:** Bousfiled-Kan spectral sequence in the context of effective homology. Complete implementation (includes central extensions, cosimplicial spaces, fiber towers, multi persistence).

3. **Formalization:** Smith reduction, persistence homology

4. **Formalizations optimization:** using modular arithmetic in formalizations to keep data size under control

5. **Generation of verified programs:** try to obtain verified and efficient programs. First example: polynomial factorization

Goals

6. **Applications:** Computational algebraic algorithms for biomedical images and network reliability

7. **Reliability:** Multi-state systems using algebraic analysis (includes filtrations, polarization and depolarization).

8. **Framework:** Integration of computer algebra algorithms in a framework for biomedical images analysis

9. **Framework extension:** Extension of this framework to incorporate and collaborate with algorithms in different contexts (includes statistics, geometry or automated learning)

The research group

- 10 researcher + 6 collaborators
- Institutions involved:
  - Universidad de La Rioja (project base)
  - Institut Fourier (France)
  - London School of Economics (UK)
  - University of Bristol (UK)

Commutative Algebra and System Reliability

A system S has n components each of which can be in several states, the system itself can be in several states.

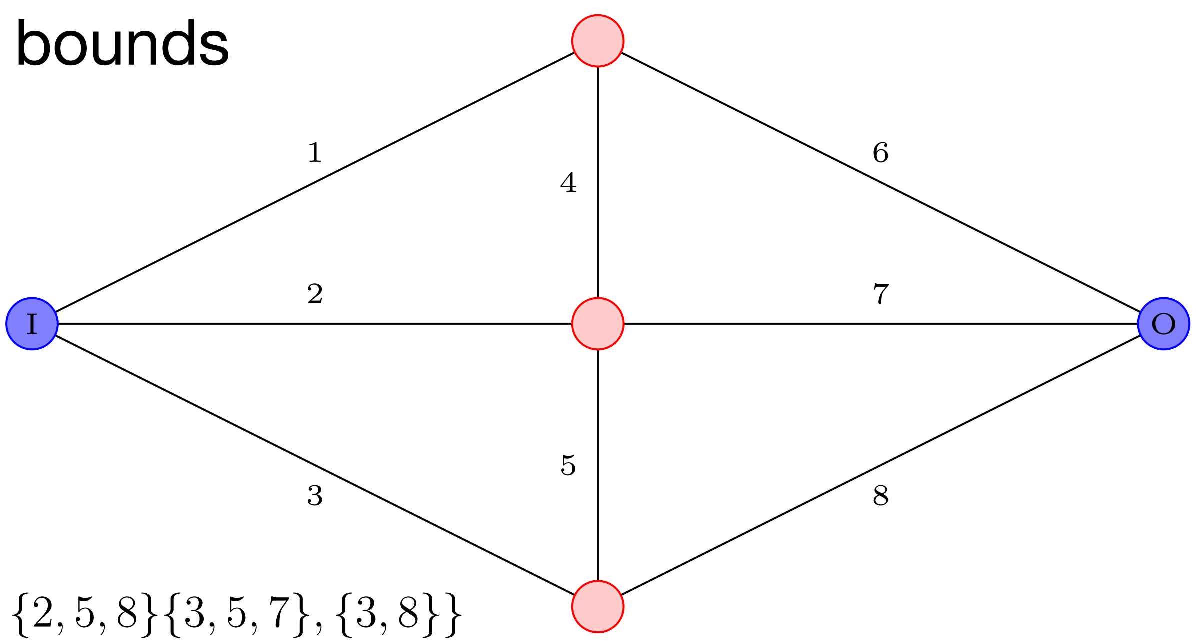
Reliability problem:

- Data:
  - probability that each component is in a given state
  - Structure function: what component states correspond to what system state
- Question:
  - Can we obtain the probability that the system is performing at level j?

Commutative Algebra and System Reliability

Algebraic approach:

- Associate a monomial ideal to each level of performance
- Obtain a minimal generating set
- Compute Hilbert series to obtain reliability polynomial
- Obtain Betti numbers to have reliability bounds



$$\{\{1,6\},\{1,4,7\},\{2,4,6\},\{1,4,5,8\},\{2,7\},\{3,4,5,6\},\{2,5,8\}\{3,5,7\},\{3,8\}\}$$

$$I_{\mathcal{S}} = \langle x_1x_6, x_1x_4x_7, x_2x_4x_6, x_1x_4x_5x_8, x_2x_7, x_3x_4x_5x_6, x_2x_5x_8, x_3x_5x_7, x_3x_8\rangle.$$

Method	Total	0	1	2	3	4	5	6	7	8
Taylor (inc-exc)	511	9	36	84	126	126	84	36	9	1
Scarf (abstr. tube)	103	9	27	37	24	6	0	0	0	0
Hilbert Series (min. res.)	87	9	25	31	18	4	0	0	0	0

Biomedical image processing

Applications in which the group is working:

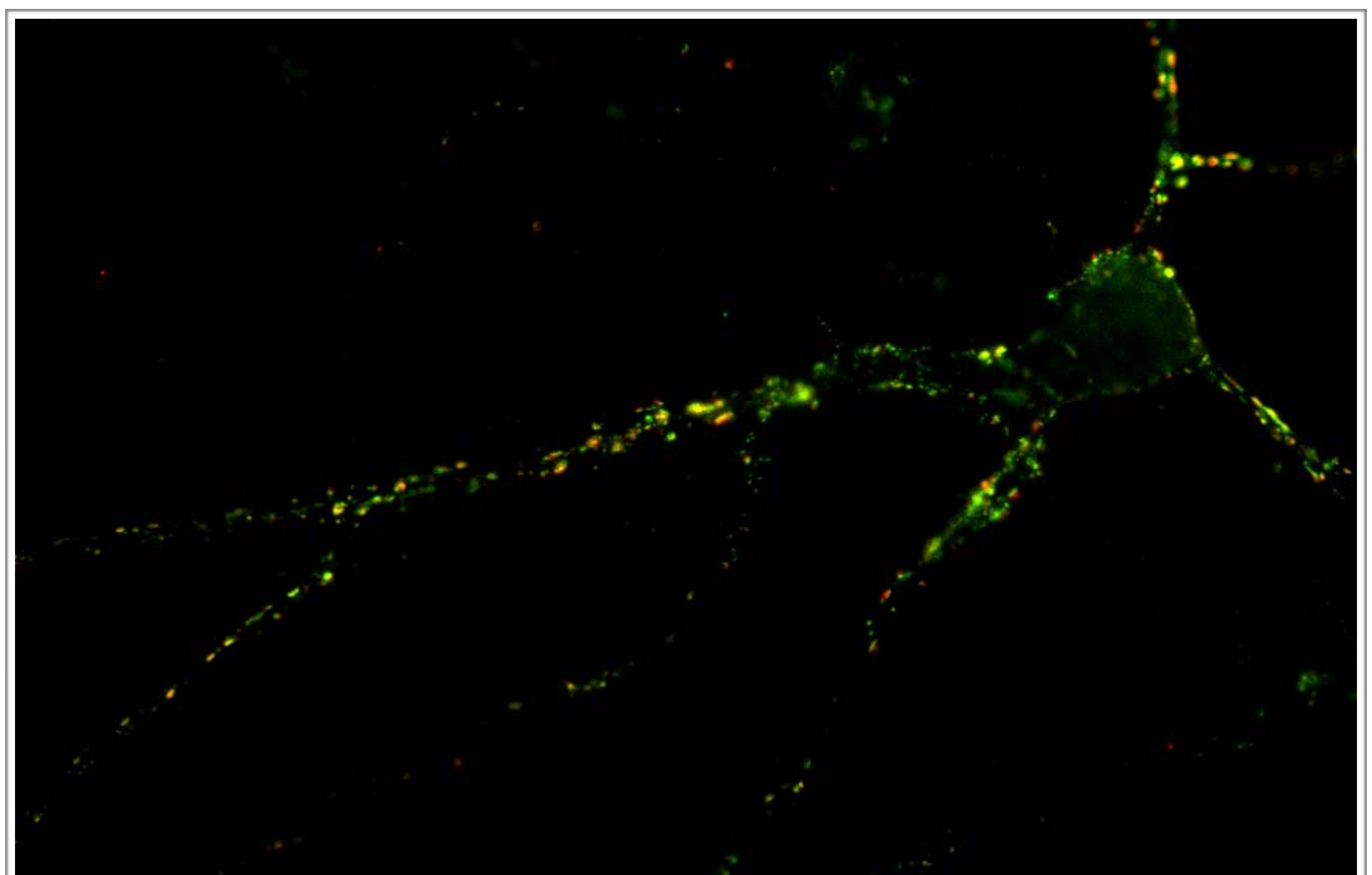
1. Synapse counting
2. Genetic footprint in DNA gel
3. Bacterial resistance to antibiotics
4. Fungi biotecnology

Synapsis counting

Biological problem: measuring synapse density in neurons

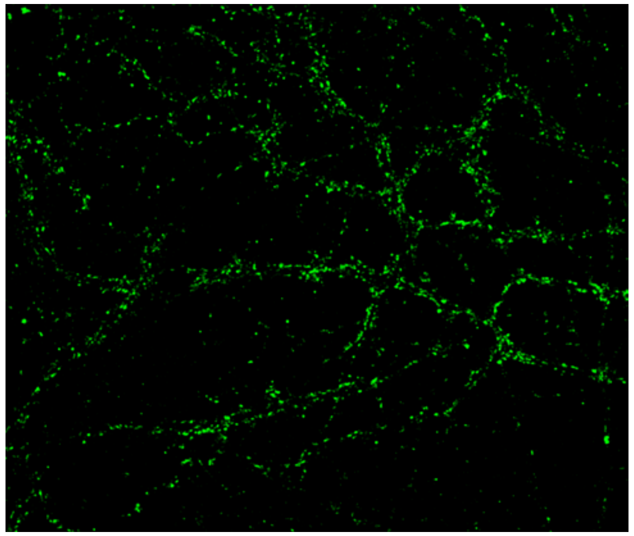
Goal: automatization of synapse counting

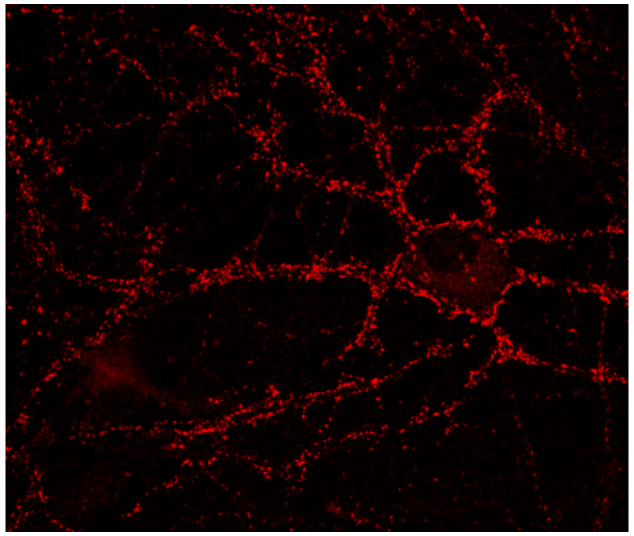
G. Mata et al. *SynapCountJ: A Validated Tool for Analyzing Synaptic Densities in Neurons.* Communications in Computer and Information Science 690:41-55, 2017.

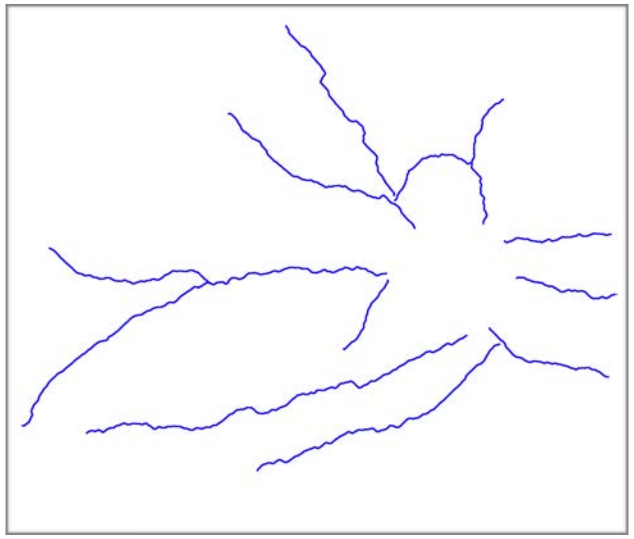


Synapsis counting

Task 1: Synapse detection

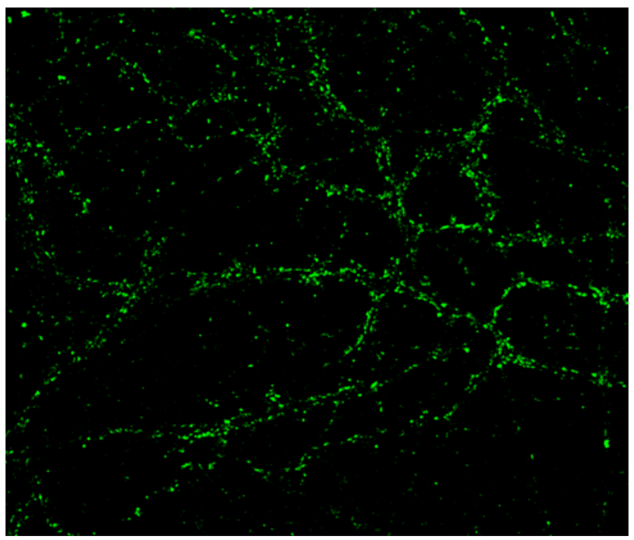


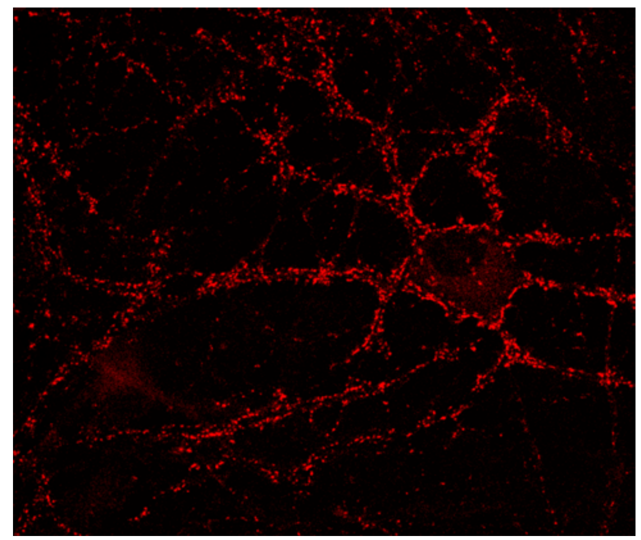


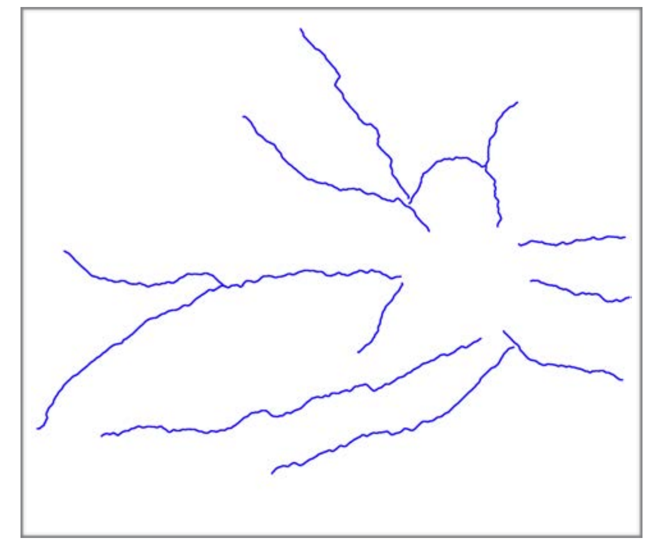


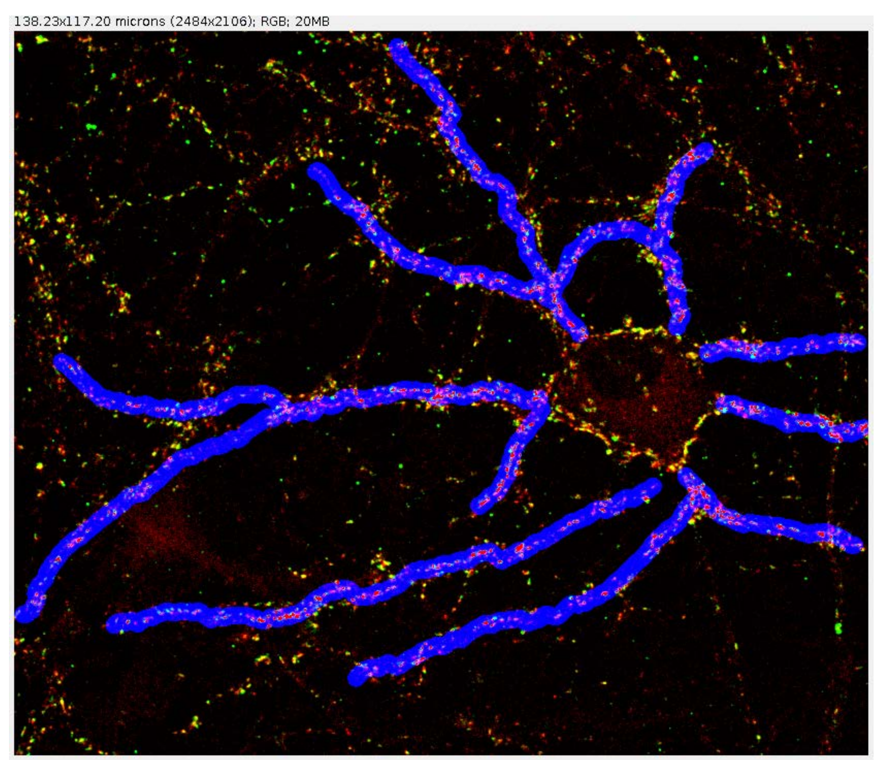
Synapsis counting

Task 1: Synapse detection



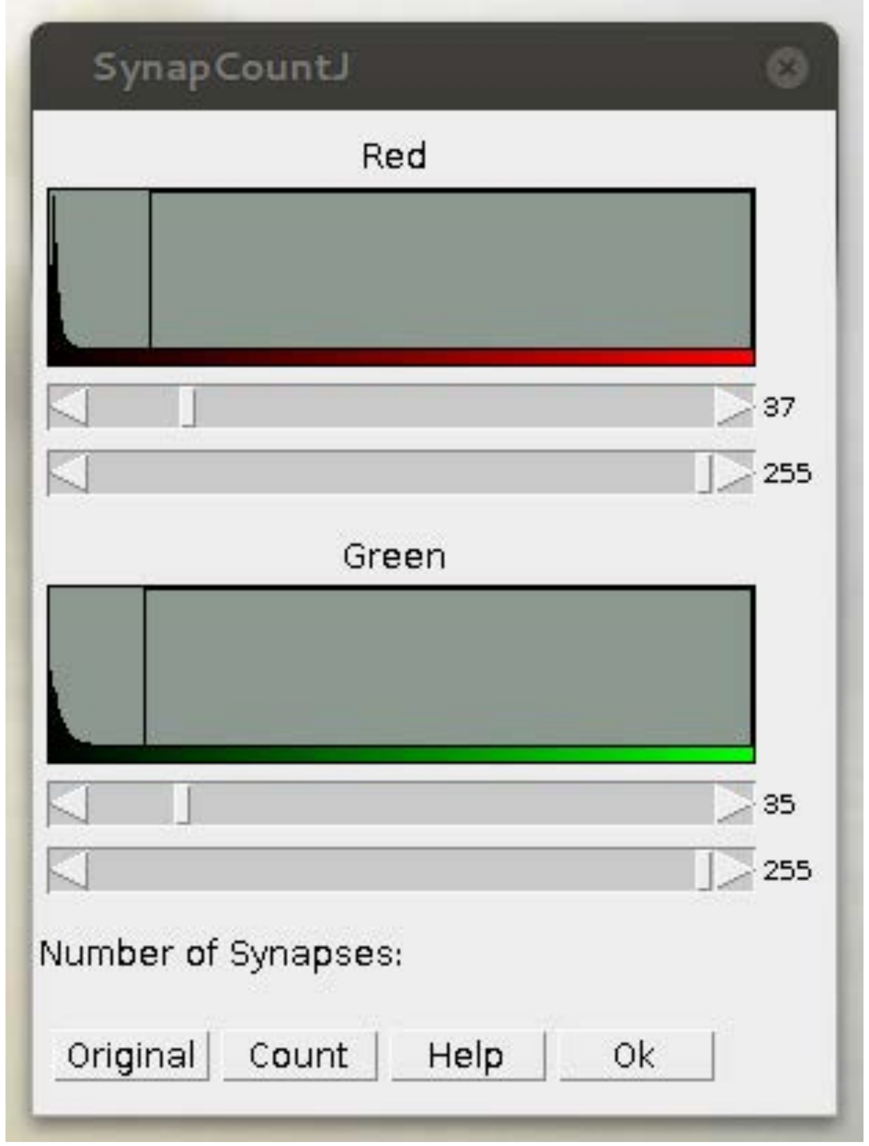


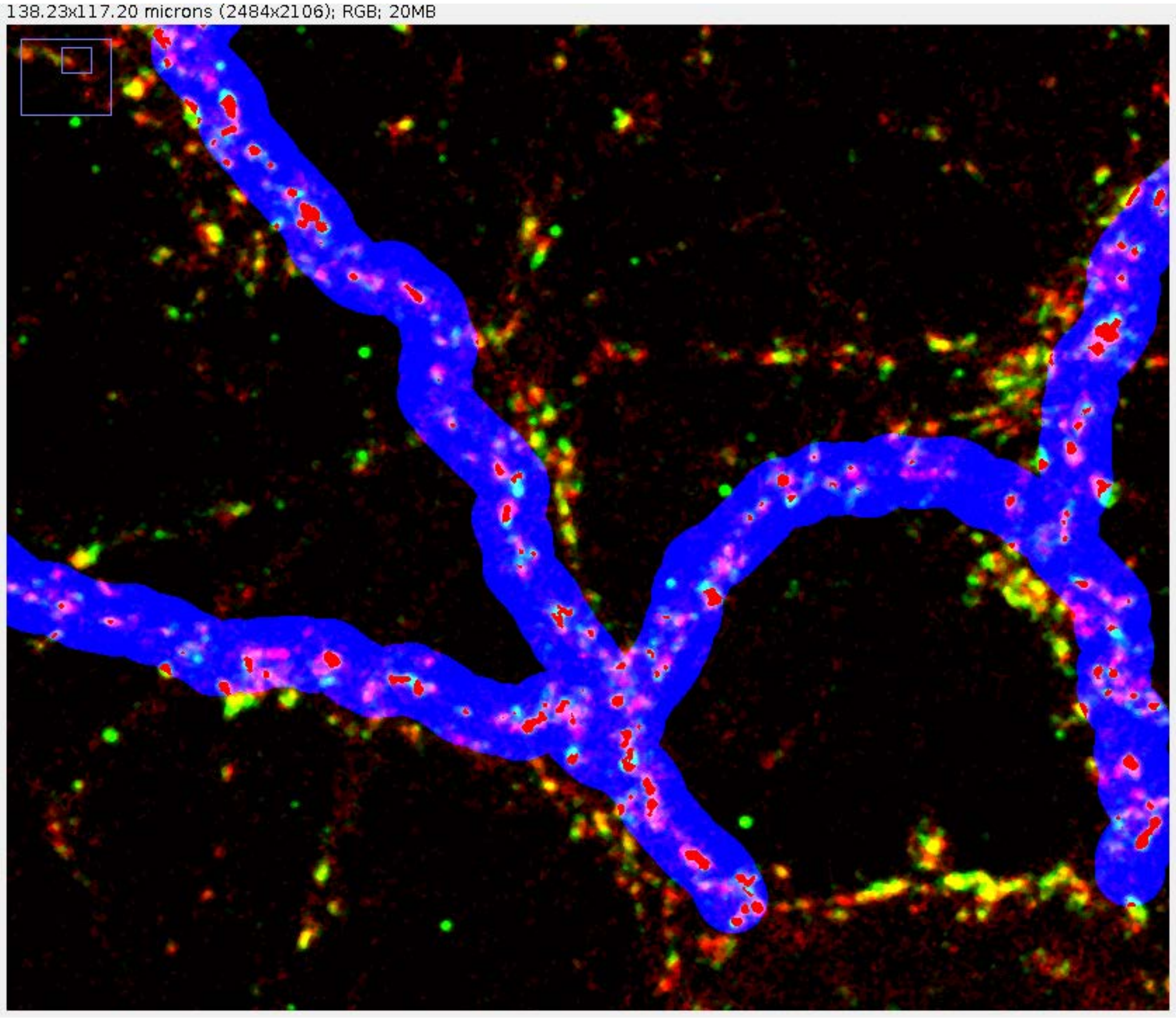




Synapsis counting

Task 2: Synapse editing



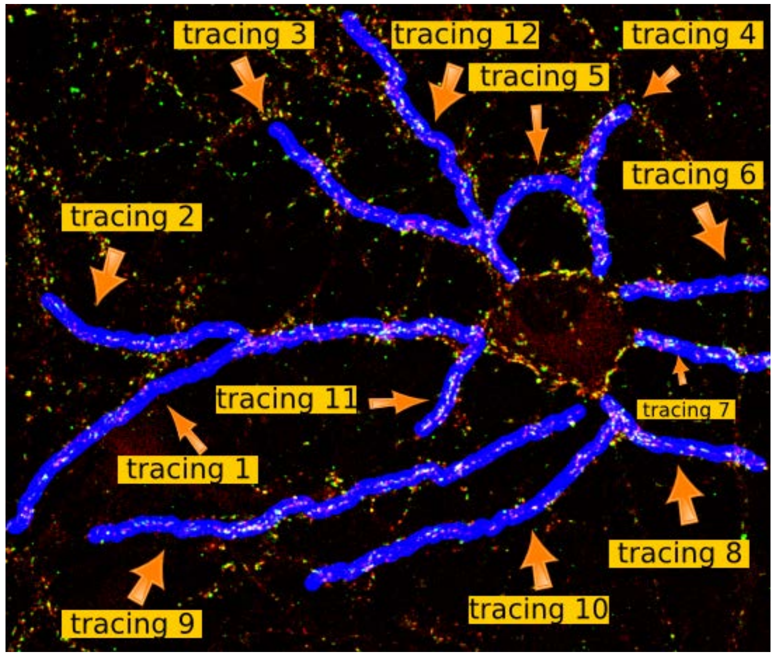


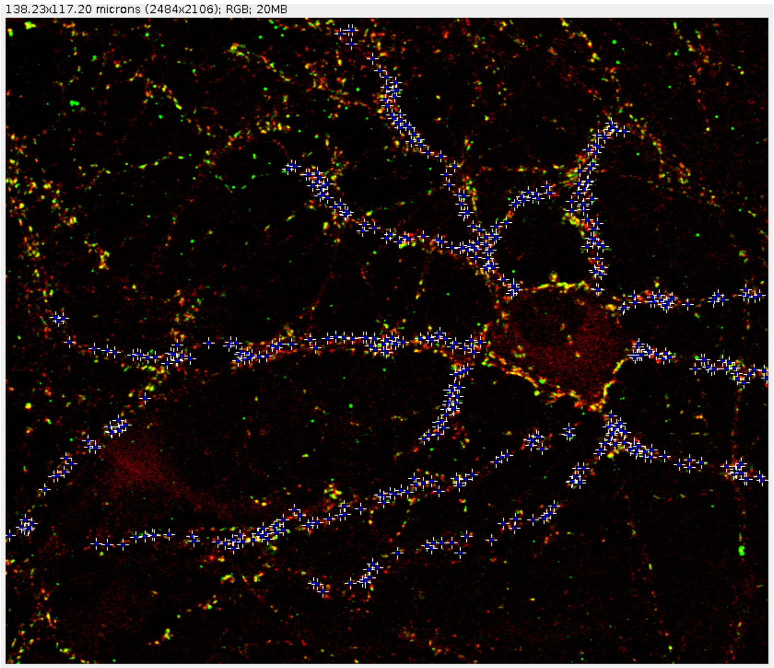
Task 3: Synapse counting

Synapsis counting

Final Result: SynapCountJ, implemented in ImageJ

	Label	Length in pixels	Length in micras	Synapses	Density	Red	Green
1	Tracing N1:	1833.1058	91.6553	71	77.4642	116	164
2	Tracing N2:	867.7840	43.3892	35	80.6652	116	164
3	Tracing N3:	983.5322	49.1766	53	107.7748	116	164
4	Tracing N4:	599.8320	29.9916	41	136.7049	116	164
5	Tracing N5:	437.7388	21.8869	25	114.2234	116	164
6	Tracing N6:	468.8438	23.4422	26	110.9111	116	164
7	Tracing N7:	447.6296	22.3815	31	138.5074	116	164
8	Tracing N8:	574.3691	28.7185	38	132.3191	116	164
9	Tracing N9:	1776.2572	88.8129	69	77.6915	116	164
10	Tracing N10:	1224.7374	61.2369	45	73.4851	116	164
11	Tracing N11:	355.7054	17.7853	26	146.1884	116	164
12	Tracing N12:	905.3750	45.2688	45	99.4063	116	164
13	Total Neuron	10474.9103	523.7455	479	91.4566	116	164





Synapsis counting

Improvement: Zig-zag homology to avoid 3D crossings

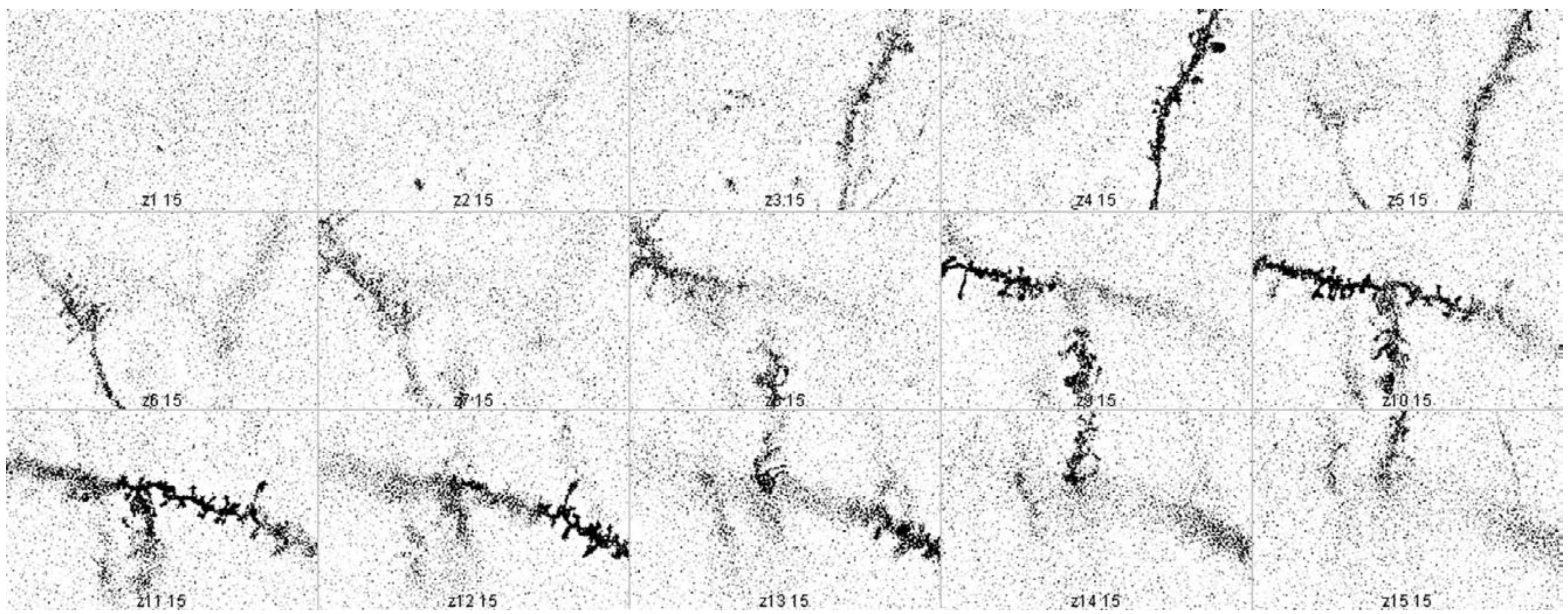


Fig. 2. A piece of Fig. 1. This is a relevant part for our study because we can see there two crossing dendrites.

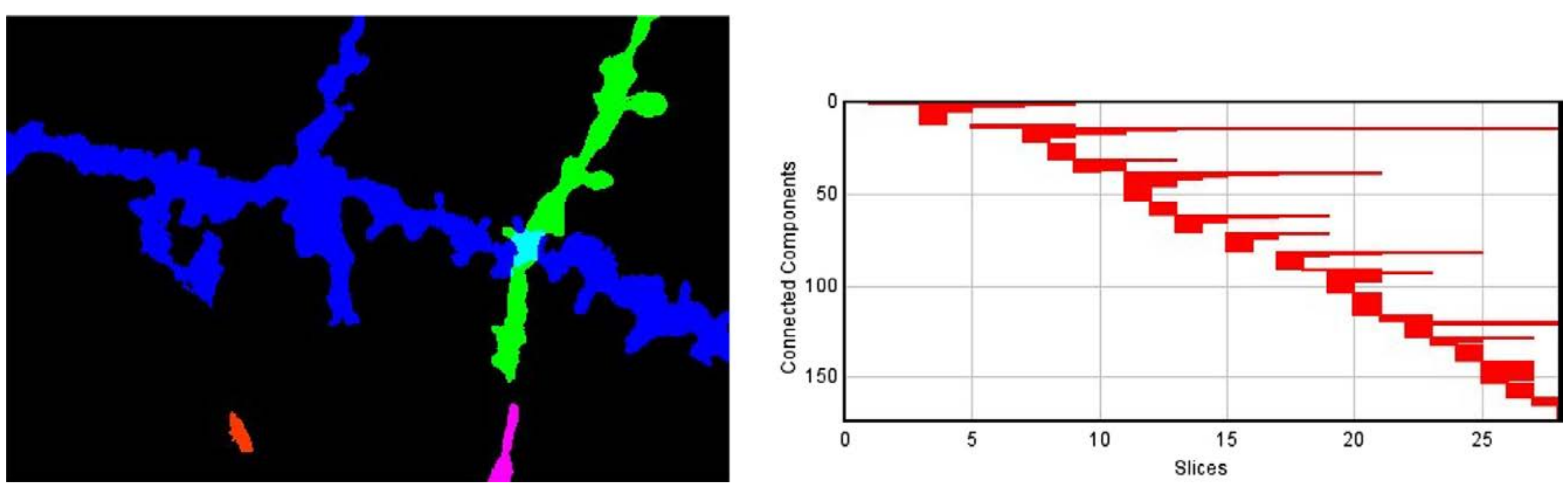


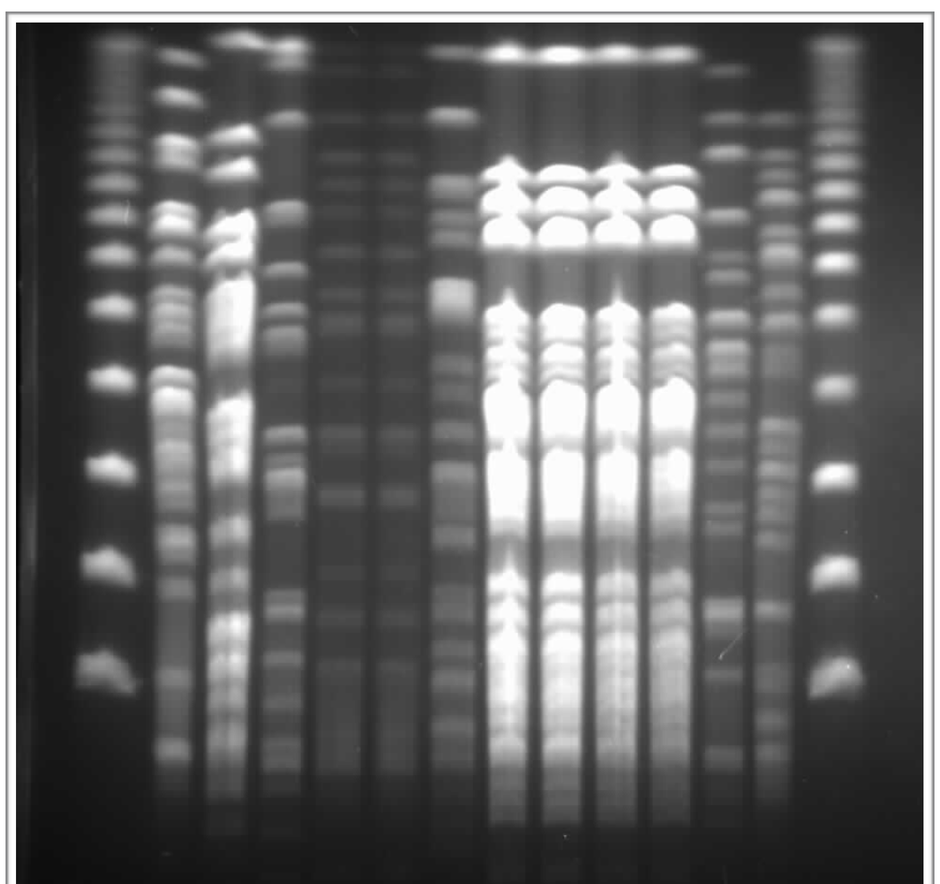
Fig. 3. Graphical results from the plugin.

Genetic footprint in DNA gel

Biological problem: compare DNA patterns

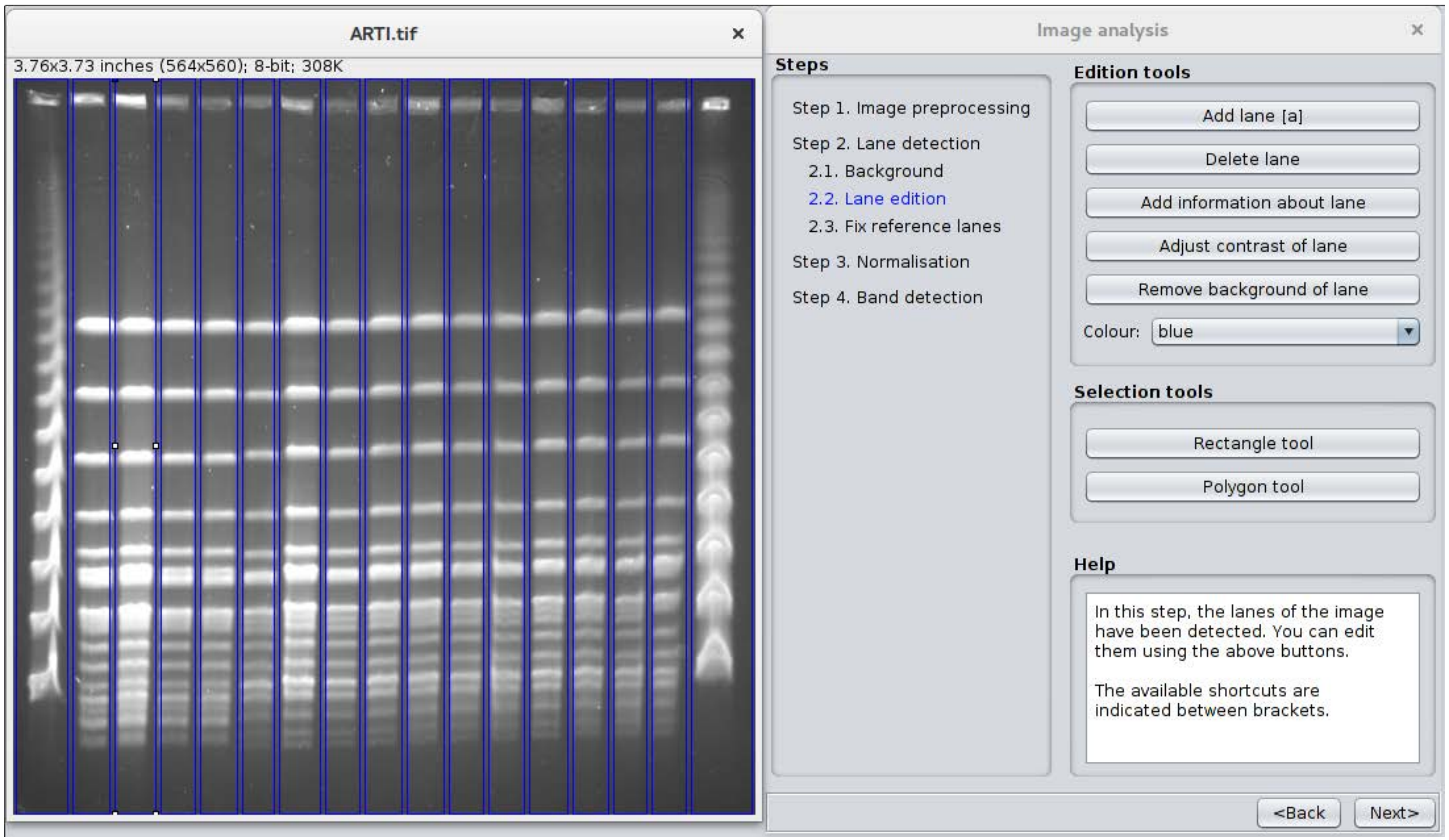
Goal: Semi-automatization of the analysis of genetic footprint gels

J. Heras et al. *GelJ -- a Tool for Analyzing DNA Fingerprint Gel Images.* BMC Bioinformatics 16(270), 2015.



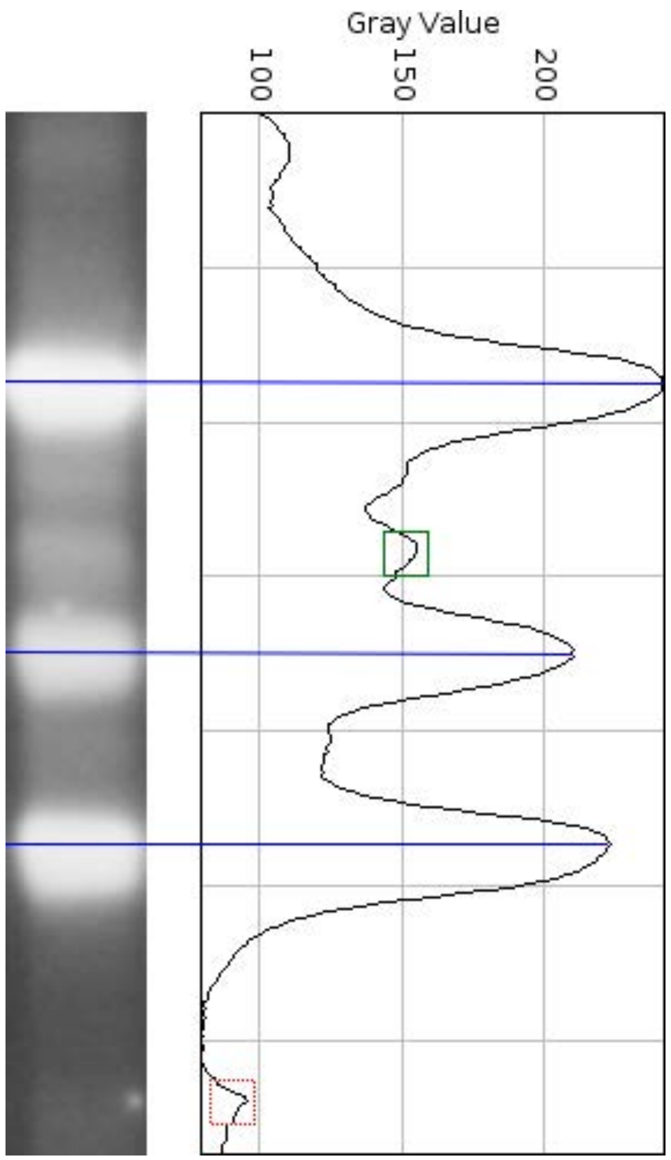
Genetic footprint in DNA gel

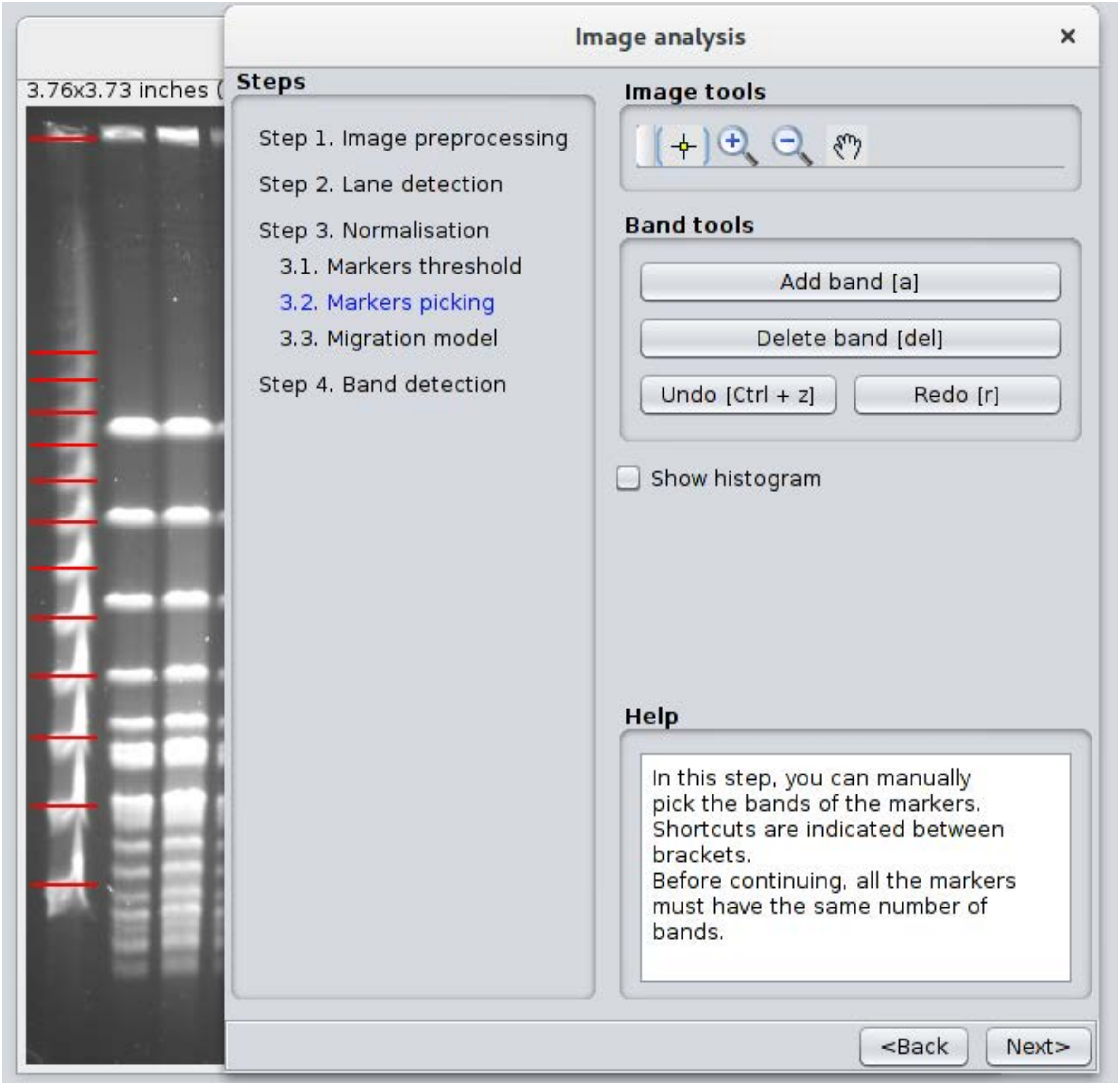
Task 1: Detect and edit rails



Genetic footprint in DNA gel

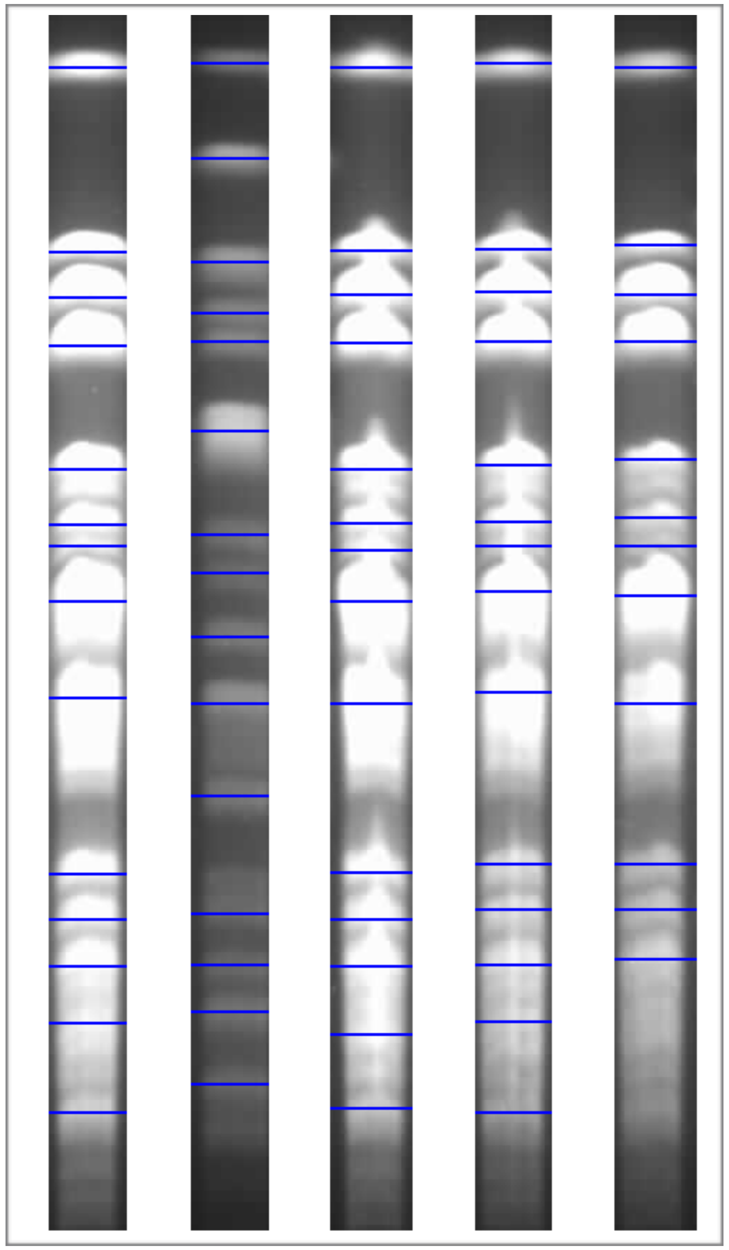
Task 2: Detect and edit strain



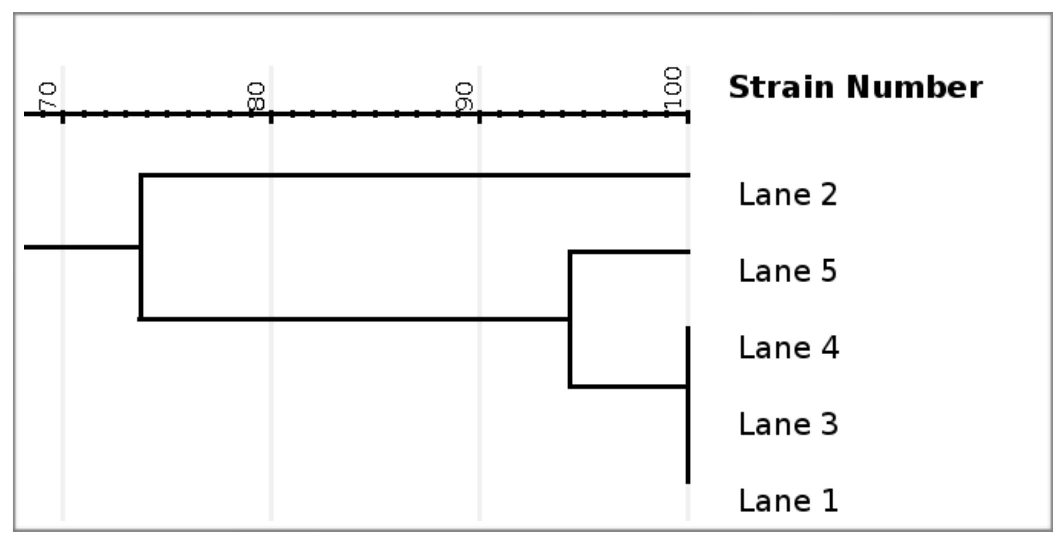


Genetic footprint in DNA gel

Task 3: Compare lanes

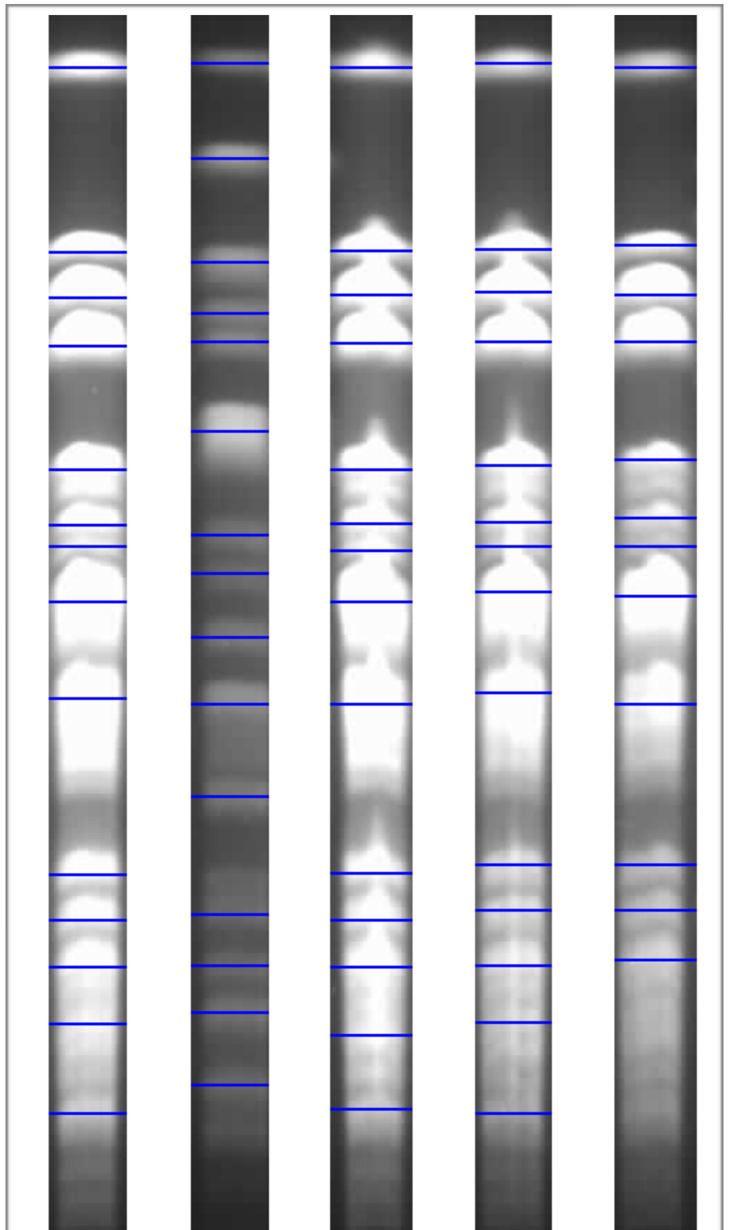


A	B	C	D	E	F
	Lane 1	Lane 2	Lane 3	Lane 4	Lane 5
Lane 1	1.0	0.66	1.0	1.0	0.92
Lane 2	0.66	1.0	0.66	0.66	0.57
Lane 3	1.0	0.66	1.0	1.0	0.92
Lane 4	1.0	0.66	1.0	1.0	0.92
Lane 5	0.92	0.57	0.92	0.92	1.0

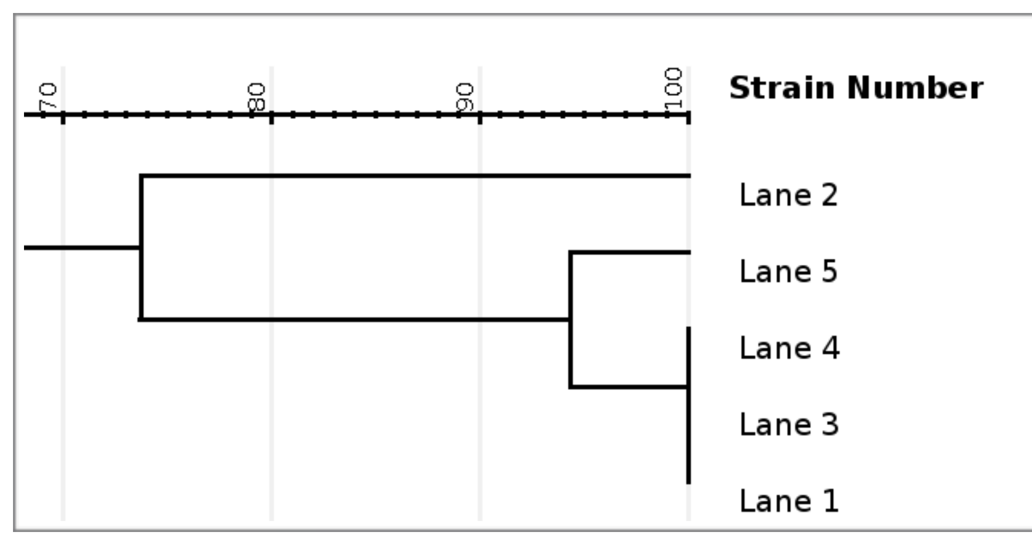


Genetic footprint in DNA gel

Final result: GelJ, free tool developed in Java combining ImageJ and Weka



A	B	C	D	E	F
	Lane 1	Lane 2	Lane 3	Lane 4	Lane 5
Lane 1	1.0	0.66	1.0	1.0	0.92
Lane 2	0.66	1.0	0.66	0.66	0.57
Lane 3	1.0	0.66	1.0	1.0	0.92
Lane 4	1.0	0.66	1.0	1.0	0.92
Lane 5	0.92	0.57	0.92	0.92	1.0

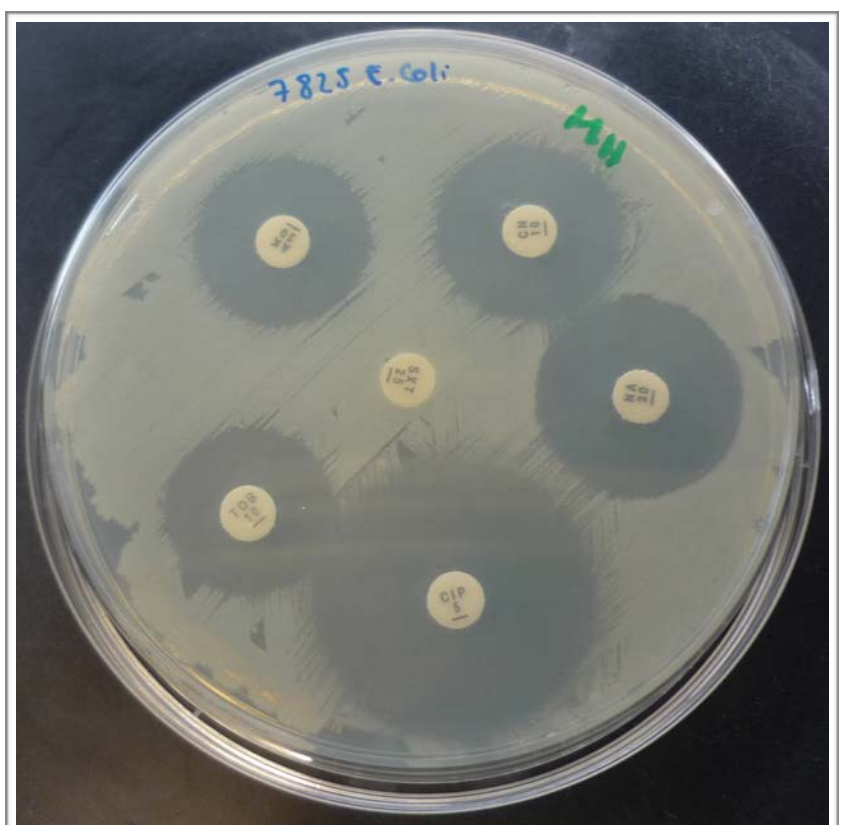


Bacterial resistance to antibiotics

Biological problem: measuring bacterial resistance to antibiotics

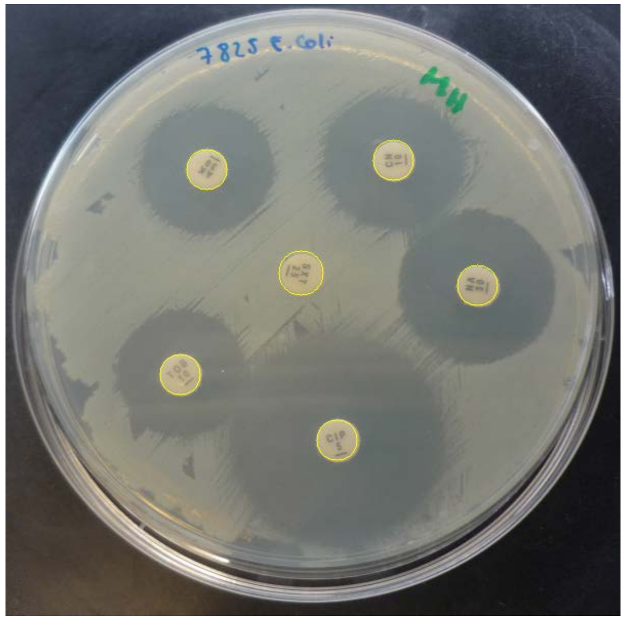
Goal: Determine, measure and categorize in an automatic or semi-automatic way zones of antibiotic inhibition,

A. Alonso et al. *AntibiogramJ: a Tool for Analysing Images in Biomedicine.* Computers in Biology and Medicine 84:189-194, 2017.



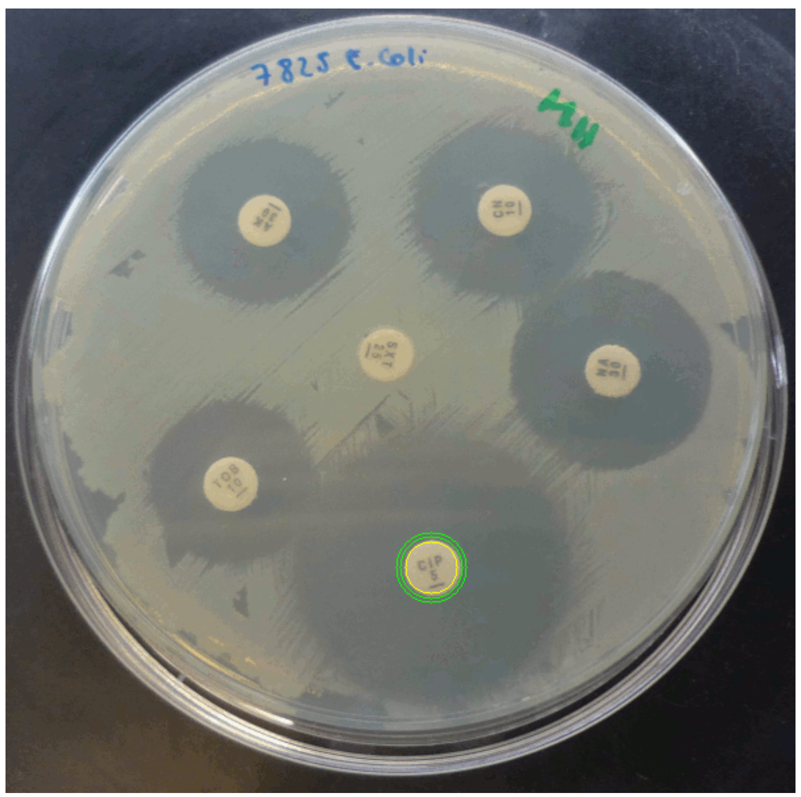
Bacterial resistance to antibiotics

Task 1: Disk detection and management



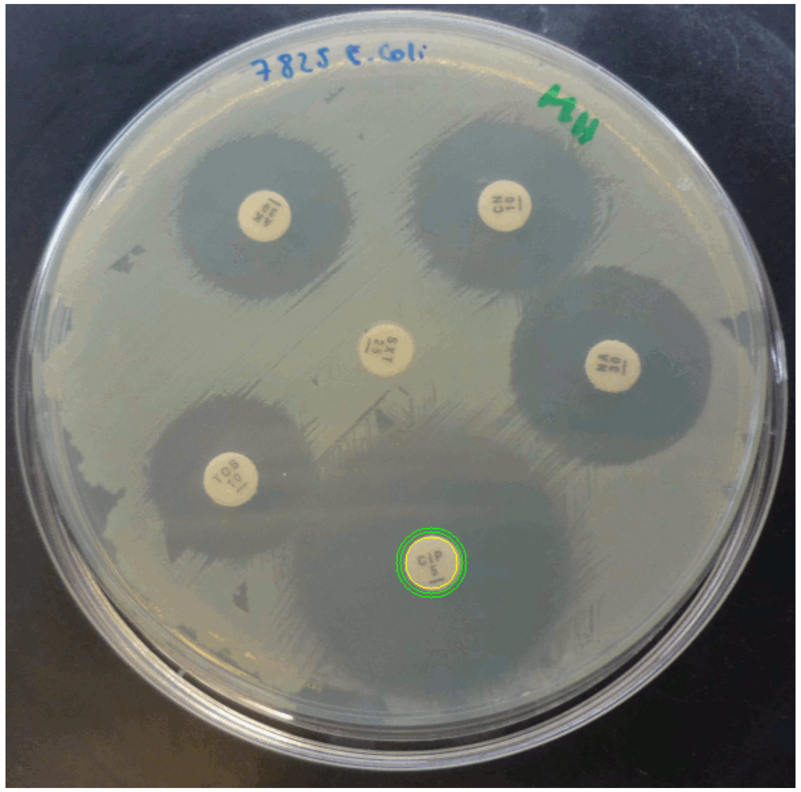
Bacterial resistance to antibiotics

Task 2: Detect, measure and categorize halos



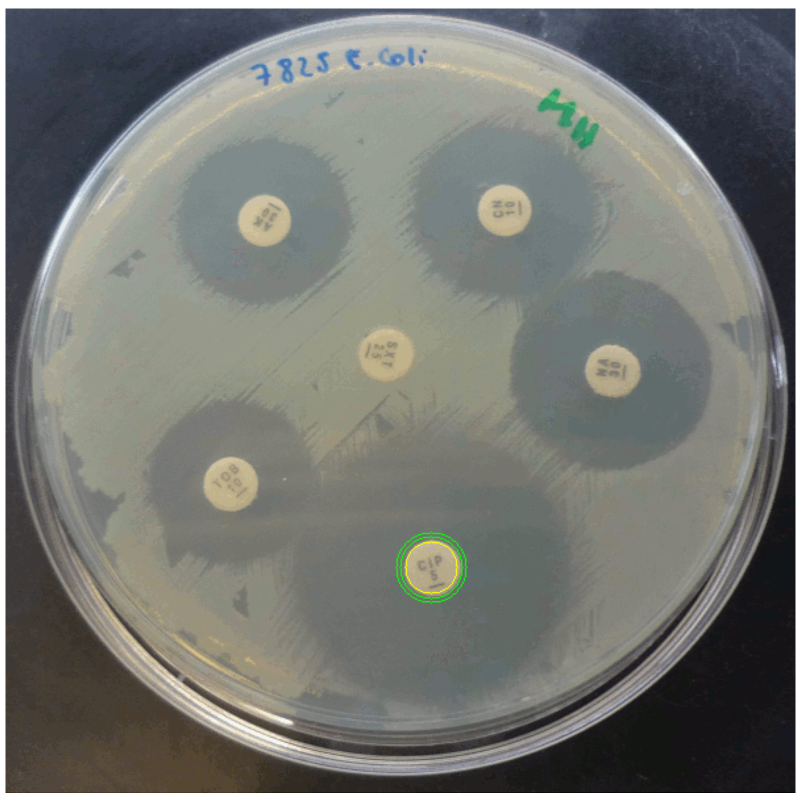
Bacterial resistance to antibiotics

Task 2: Detect, measure and categorize halos



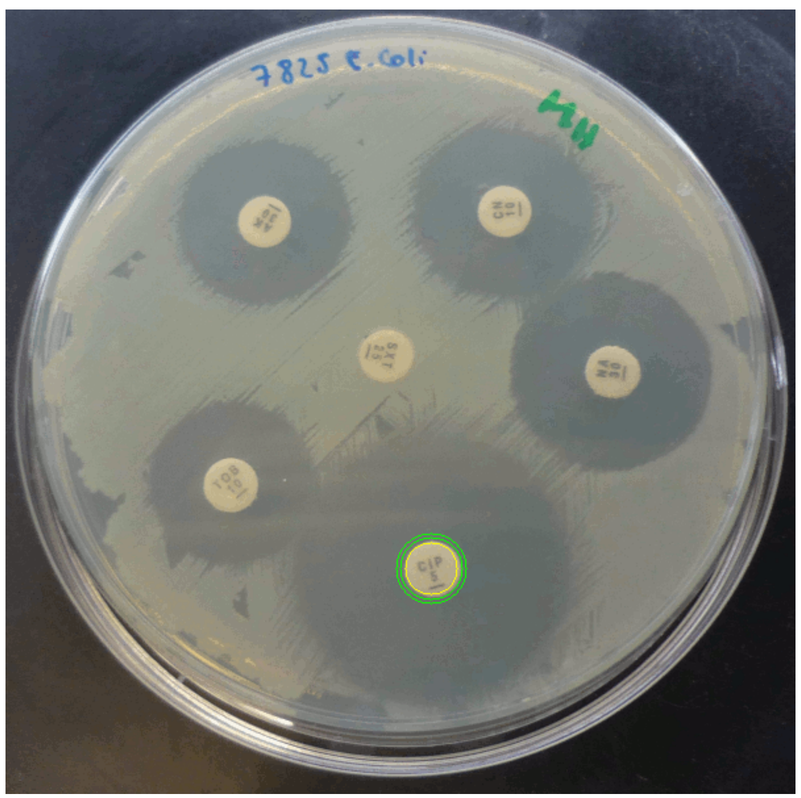
Bacterial resistance to antibiotics

Final result: AntibiogramJ, free tool developed in Java combining ImageJ and OpenCV



Bacterial resistance to antibiotics

Final result: AntibiogramJ, free tool developed in Java combining ImageJ and OpenCV



Computer algebra: Formalization and applications to network reliability and biomedical image processing

Guanajuato, Nov 20th 2018

Eduardo Sáenz de Cabezón

Universidad de La Rioja (Spain)

eduardo.saenz-de-cabezon@unirioja.es

UNIVERSIDAD DE LA RIOJA