



cLEsperanto

A GPU-accelerated image processing framework across languages and platforms

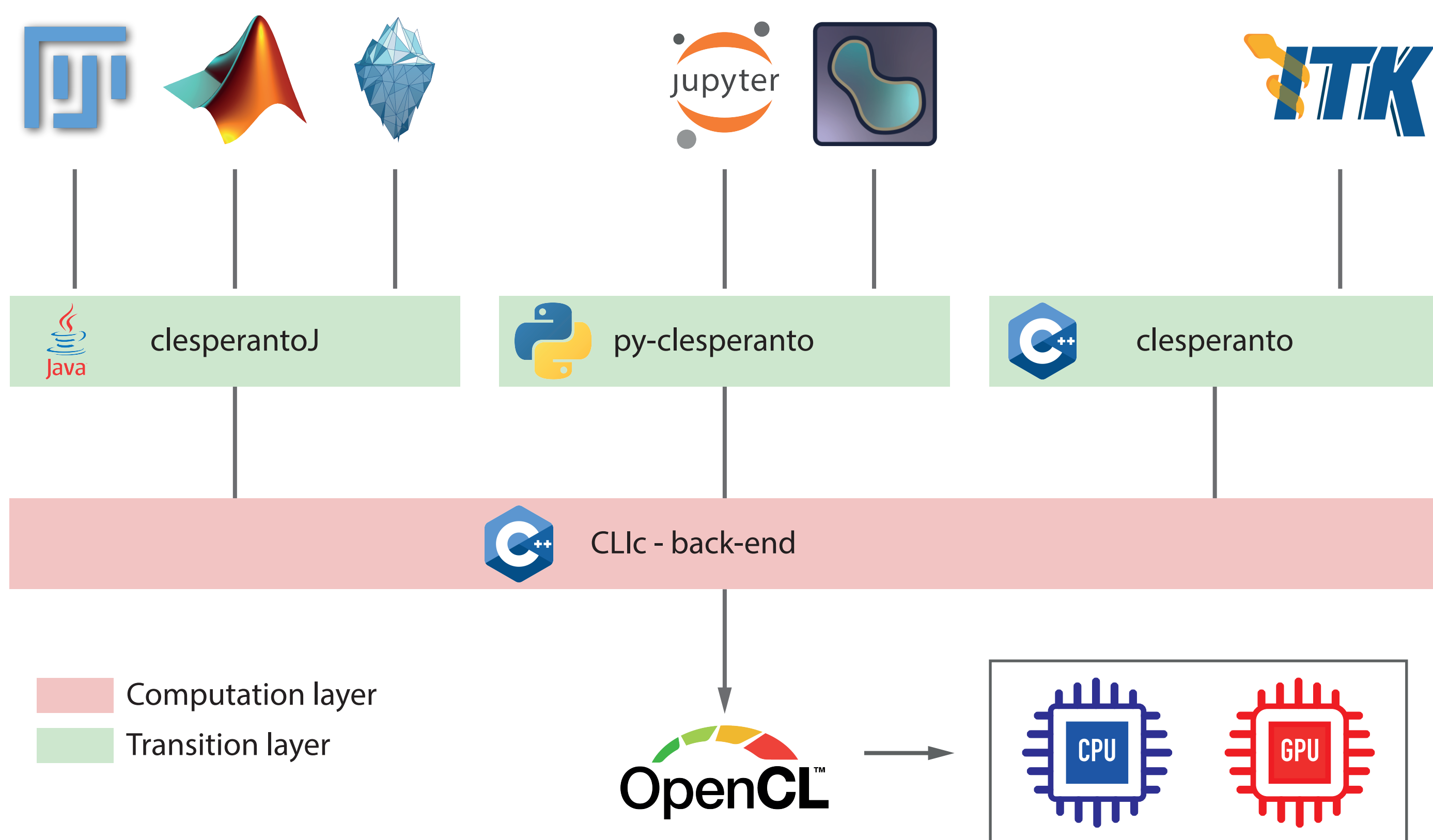
Stéphane Rigaud¹, Robert Haase²

¹ Institut Pasteur, Université de Paris Cité, Image Analysis Hub, F-75015 Paris, France

² Physics of Life Cluster of Excellence, Technische Universität, Dresden

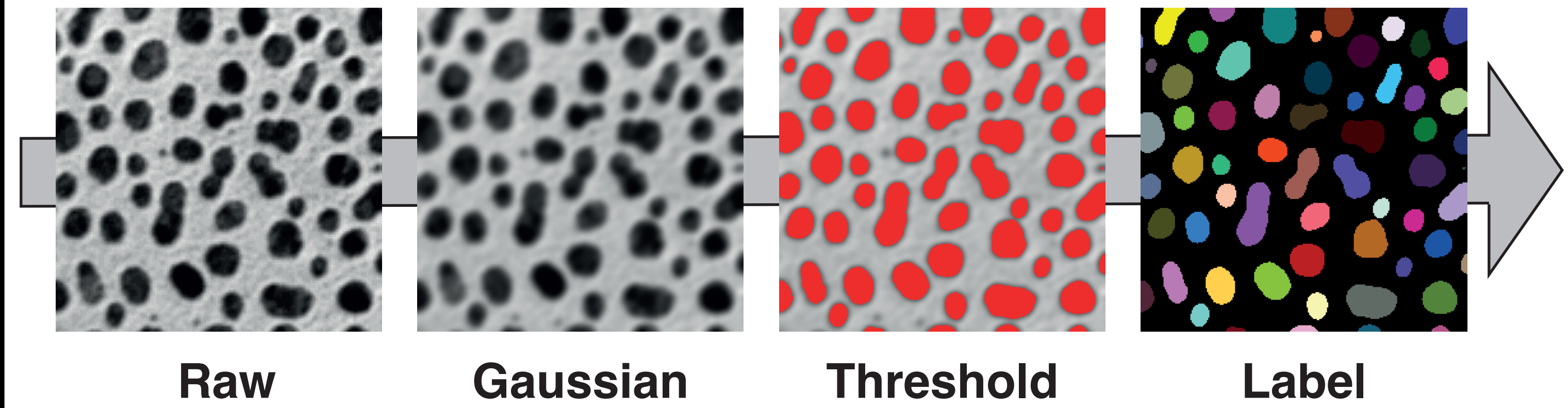
cLEsperanto is a collaborative open-source project, built on the evolution of the CLIJ platform, that aims to facilitate end-user access to modern computing hardware through an abstraction layer. Relying on a C++ low-level back-end structure, this layer allows calling GPU-accelerated image processing operations without the need for learning any specific complex language. Through this new architecture, the library is available for all major Bio-Image analysis softwares and languages while sharing the same functionality and syntax, and running the same implementation.

A unified architecture library



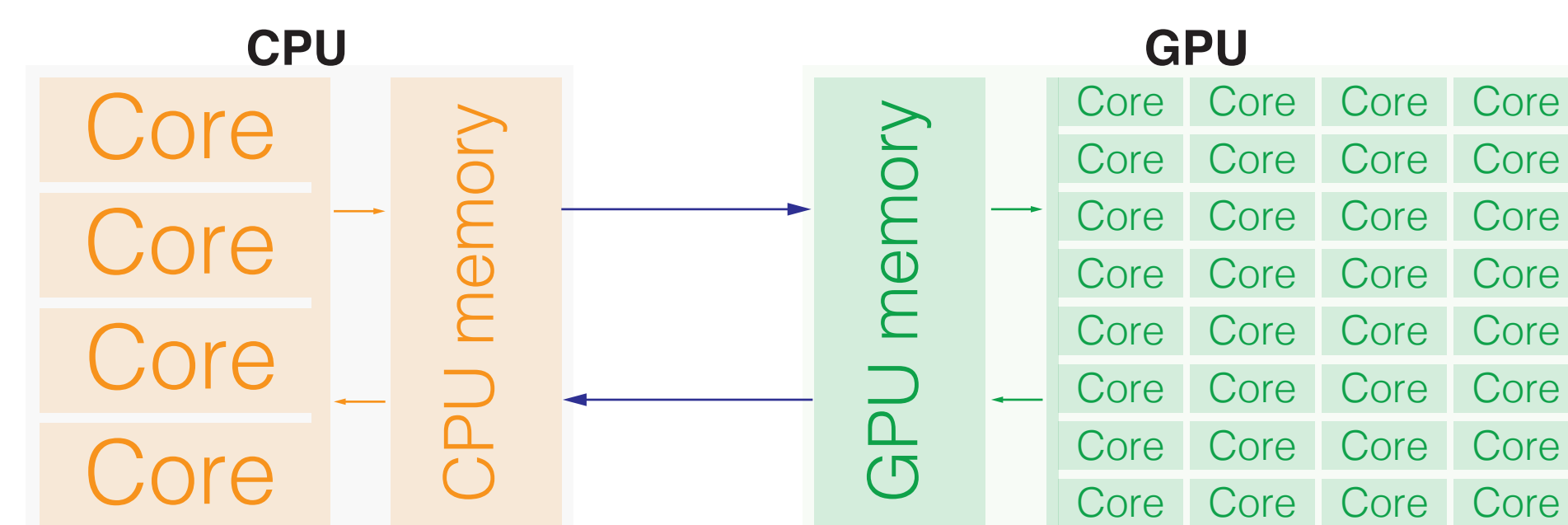
- Low-level computational layer
- Language specific wrapper API
- Target bio-image analysis environment
- GPU-acceleration using open standard OpenCL
- Open-source collaborative project

Bio-image Analysis workflows



- Redundancy of implementation and maintenance
- Absence of standardisation of name and usage
- Computational inconsistency between frameworks
- Cumulated technology debt limiting main frameworks

Processing Units



- **CPU** : few generic processing units
- **GPU** : many specialised processing units

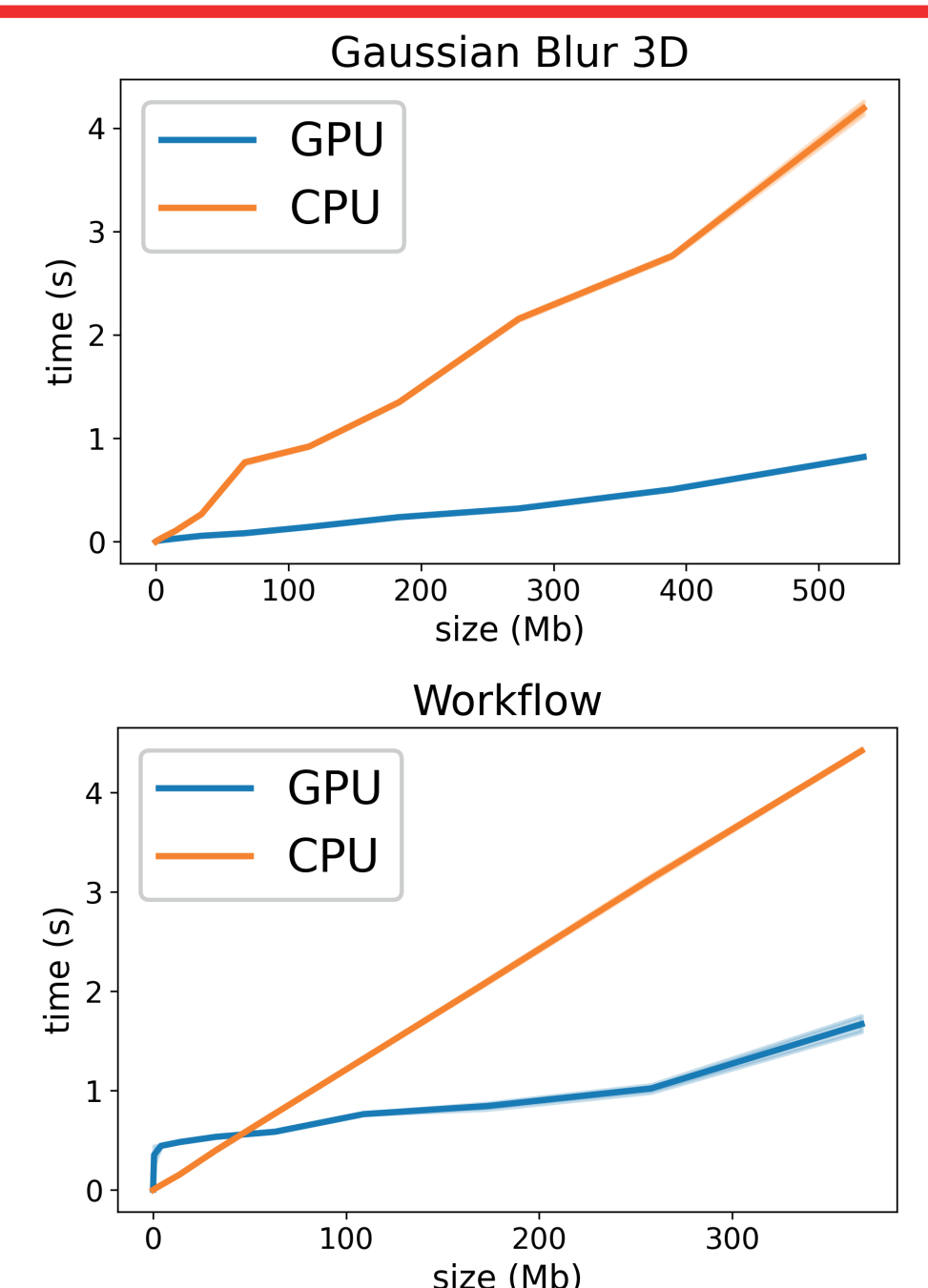
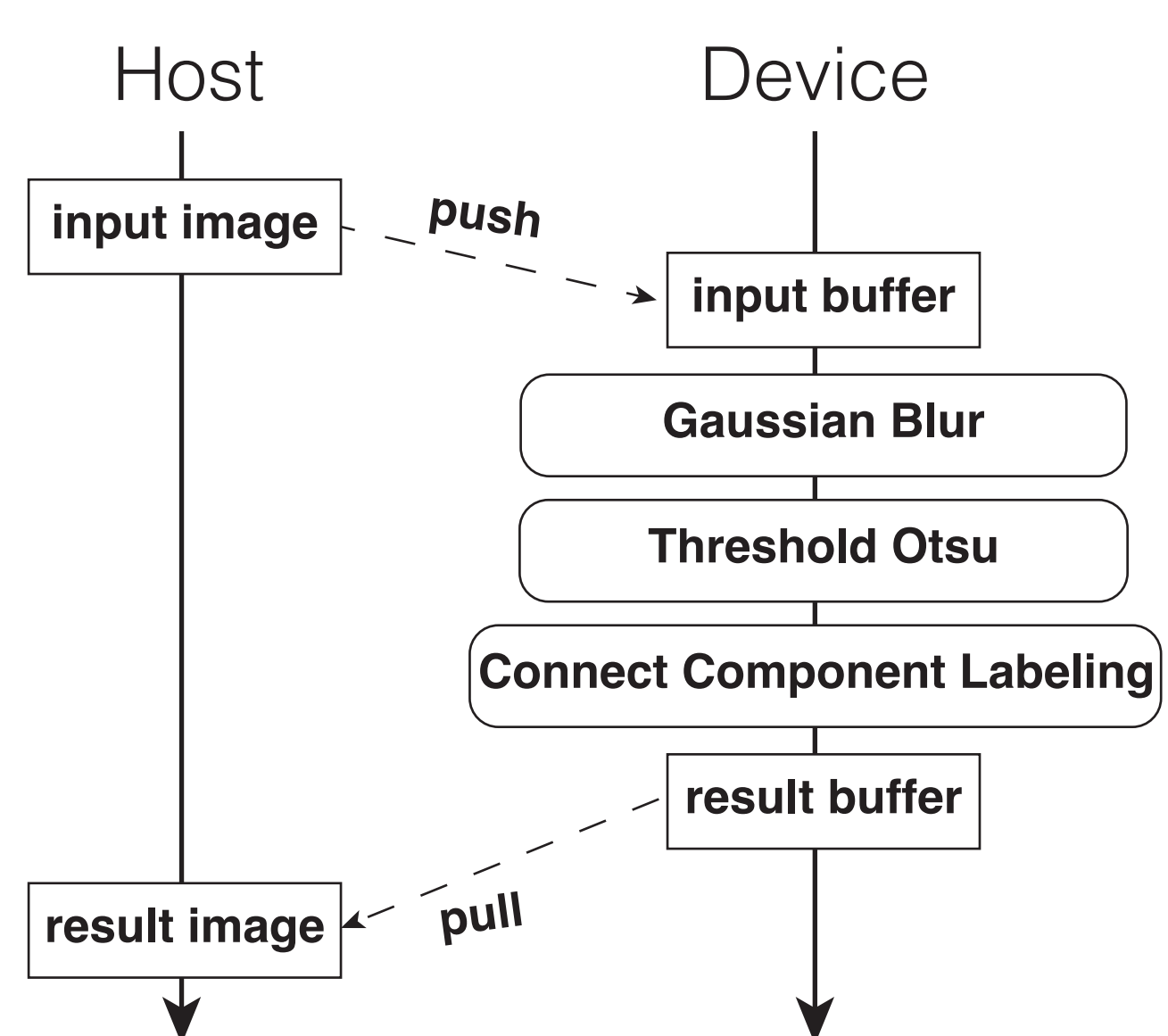


Image processing with cLEsperanto



- Pipeline oriented
- Cross-framework compatibility with minimal changes
- Clear and common operations name
- Standardised API usage

`cle.my_operation(input, output, args)`

```
#include "clesperanto.hpp"

auto input = cle.Push(image, shape);

auto blurred = cle.Create(input);
cle.GaussianBlur(input, blurred, 3, 3);

auto binary = cle.Create(blurred);
cle.ThresholdOtsu(blurred, binary);

auto label = cle.Create(binary);
cle.ConnectComponentLabeling(binary, label);

auto output = cle.Pull(label);
```

.cpp

```
import clesperanto as cle

input = cle.push(image)

blurred = cle.create_like(input)
cle.gaussian_blur(input, blurred, 3, 3)

binary = cle.create_like(blurred)
cle.threshold_otsu(blurred, binary)

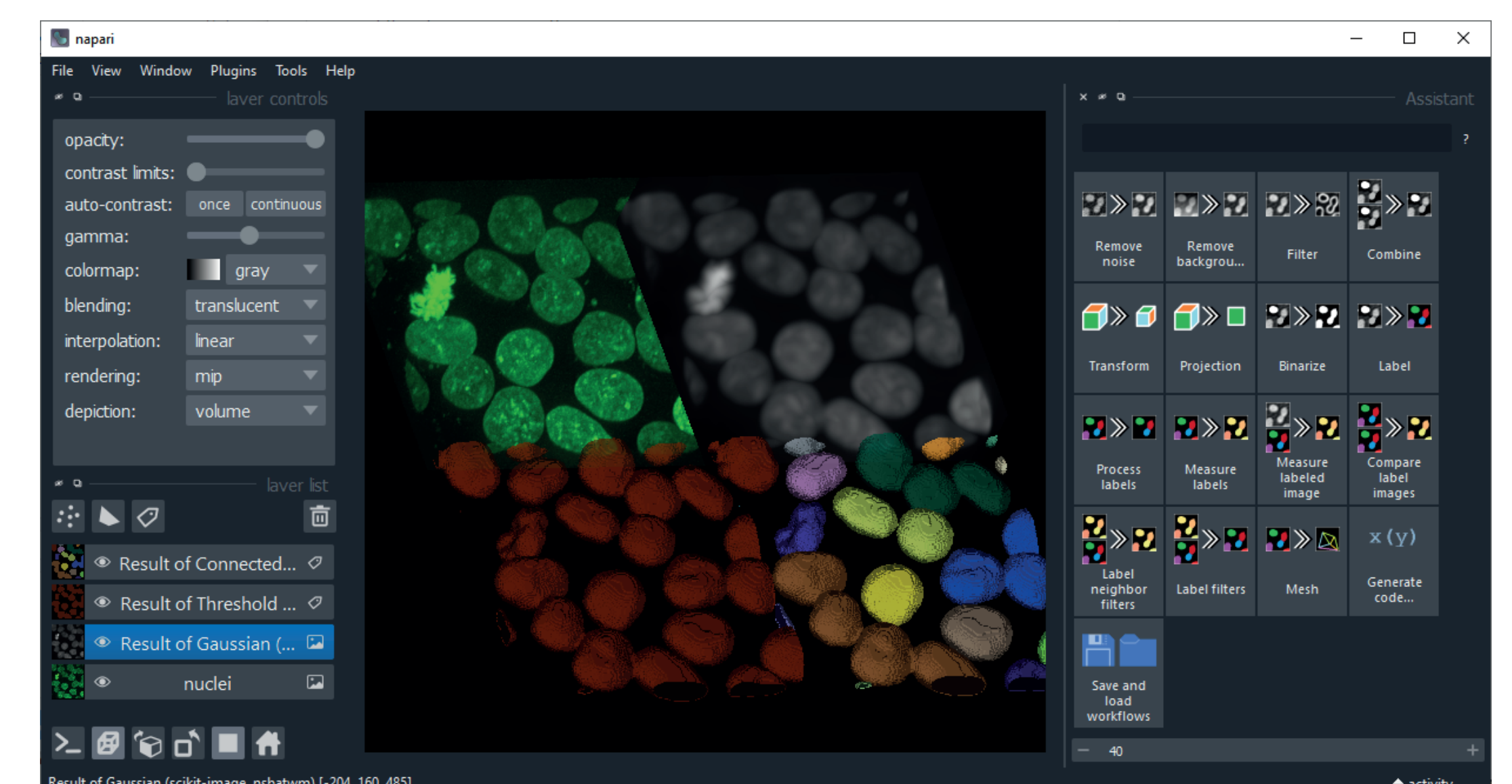
label = cle.create_like(binary)
cle.connect_component_labeling(binary, label)

output = cle.pull(label)
```

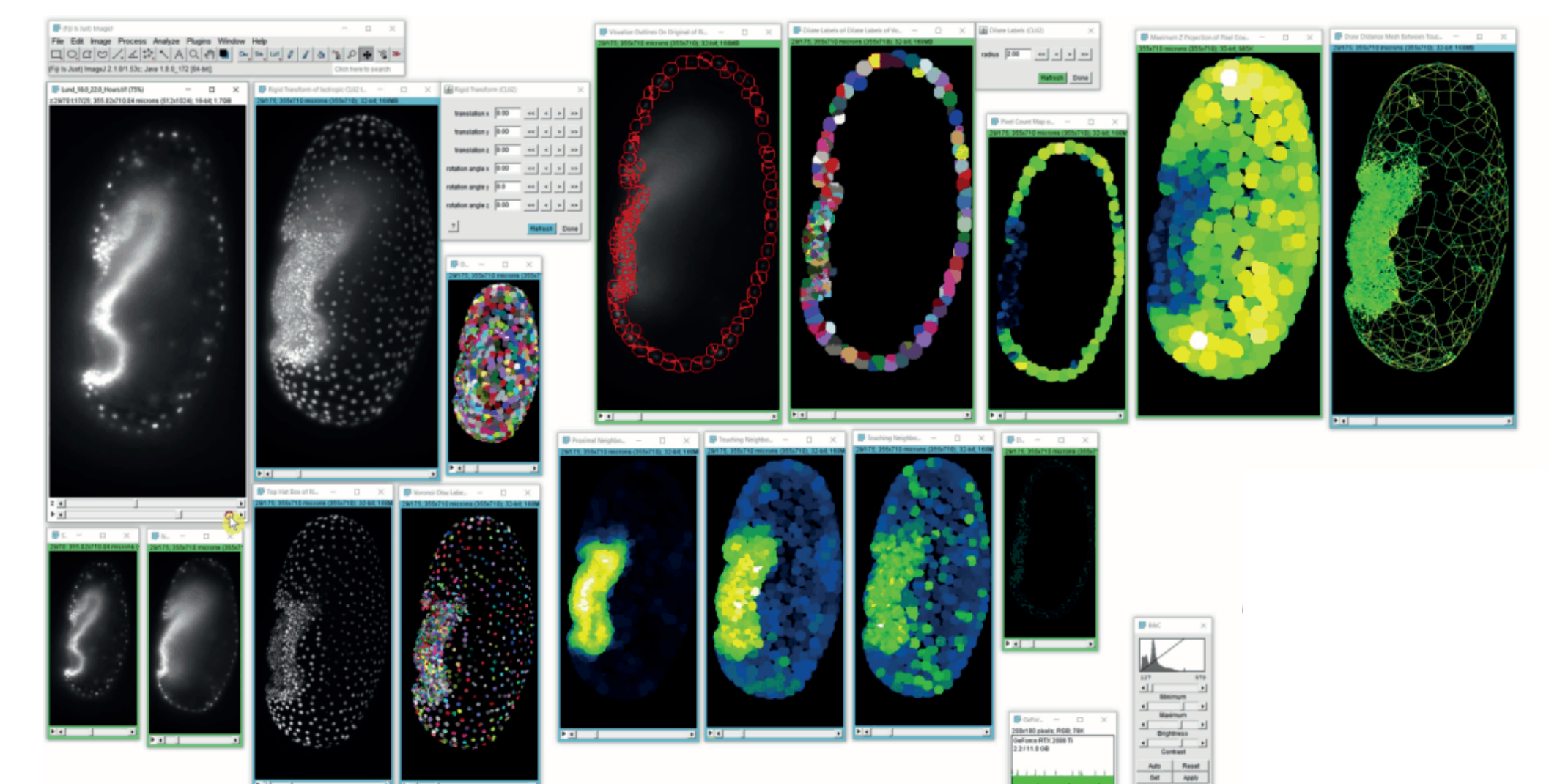
.py

cLEsperanto Assistants

Napari Assistant



Fiji Assistant



- GUI image processing workflow builder
- Remove programming language barriers
- Script export for supported frameworks

References:

- Robert Haase, Loic A. Royer, et. al. *CLIJ: GPU-accelerated image processing for everyone*, Nature Methods, 2020
- Robert Haase, Akanksha Jain, Stéphane Rigaud, et. al. *Interactive design of GPU-accelerated Image Data Flow Graph and cross-platform deployment using multi-lingual code generation*, BiorXiv, 2020
- Robert Haase, Talley Lambert, Justin Kiggins, Johannes Müller, & Kevin Yamauchi. (2022). cLEsperanto/napari_pyclesperanto_assistant: 0.21.0. Zenodo.

