

# hist<sup>3</sup>lab: an Open Source Python Library for Reproducible Digital Pathology

Alessia Marcolini

[alessia.marcolini@hk3lab.ai](mailto:alessia.marcolini@hk3lab.ai)

 @viperale

Ernesto Arbitrio

[ernesto.arbitrio@gmail.com](mailto:ernesto.arbitrio@gmail.com)

 @\_pamaron\_



*Online Tech Conference*

- Italian edition -

C/C++



C/C++



9-10-11 Novembre, 2021

# Alessia

❖ Data Science M.Sc. @ TU Eindhoven / TU Berlin

❖ Junior Data Scientist @ HK3lab

❖ PyCon Italia Organizer

# Ernesto

❖ Senior Backend Engineer @ YouGov PLC

❖ PyCon Italia Organizer

❖ Open Source Contributor



day 1

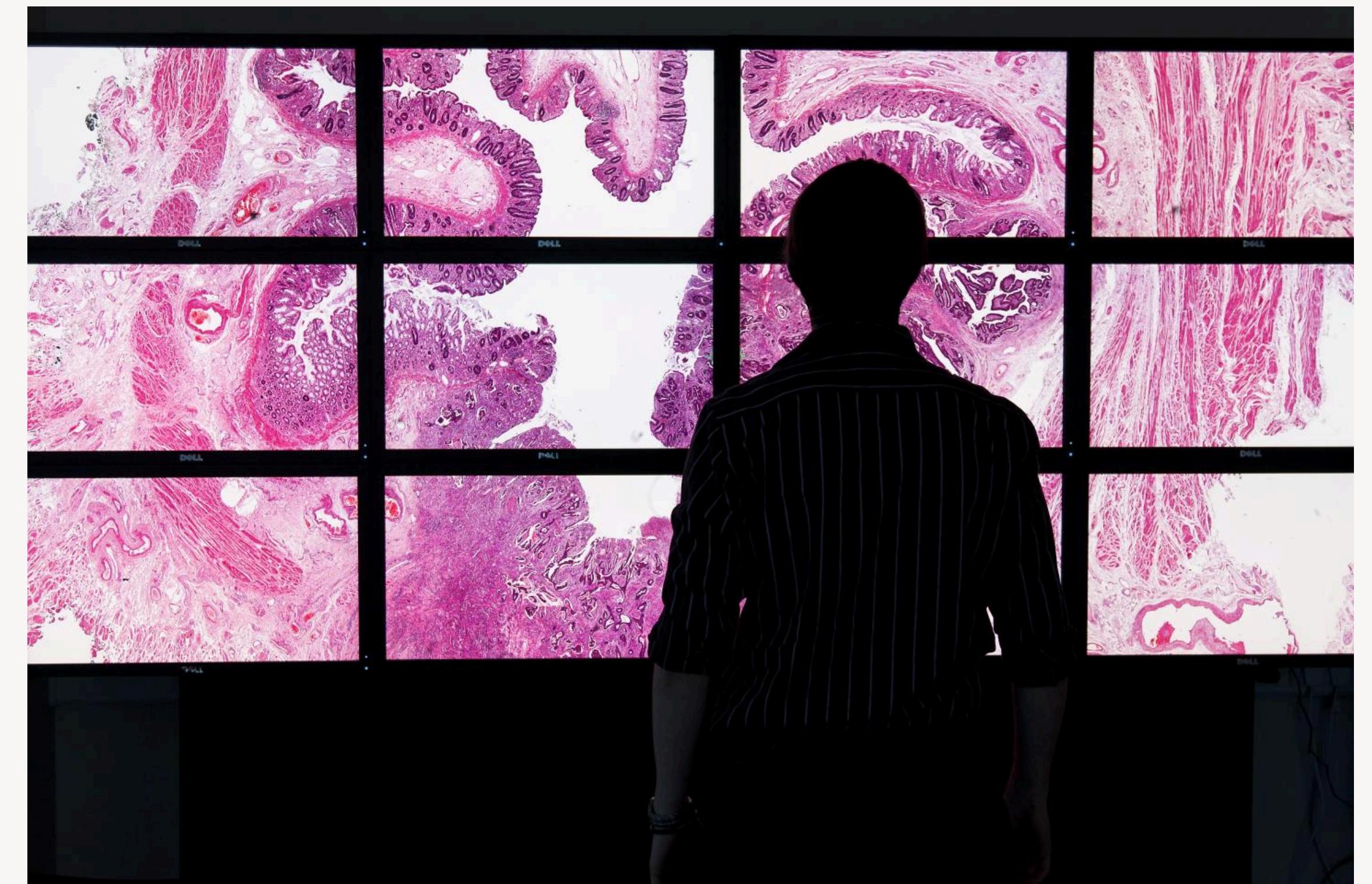
# Histopathology



<https://www.poliambulanza.it/dipartimenti/dipartimento-di-oncologia/anatomia-patologica>

**Primary diagnostic resource for  
the identification of complex  
diseases, in particular of tumors**

**Scanning of histopathological glass  
slides to create Whole Slide Images**

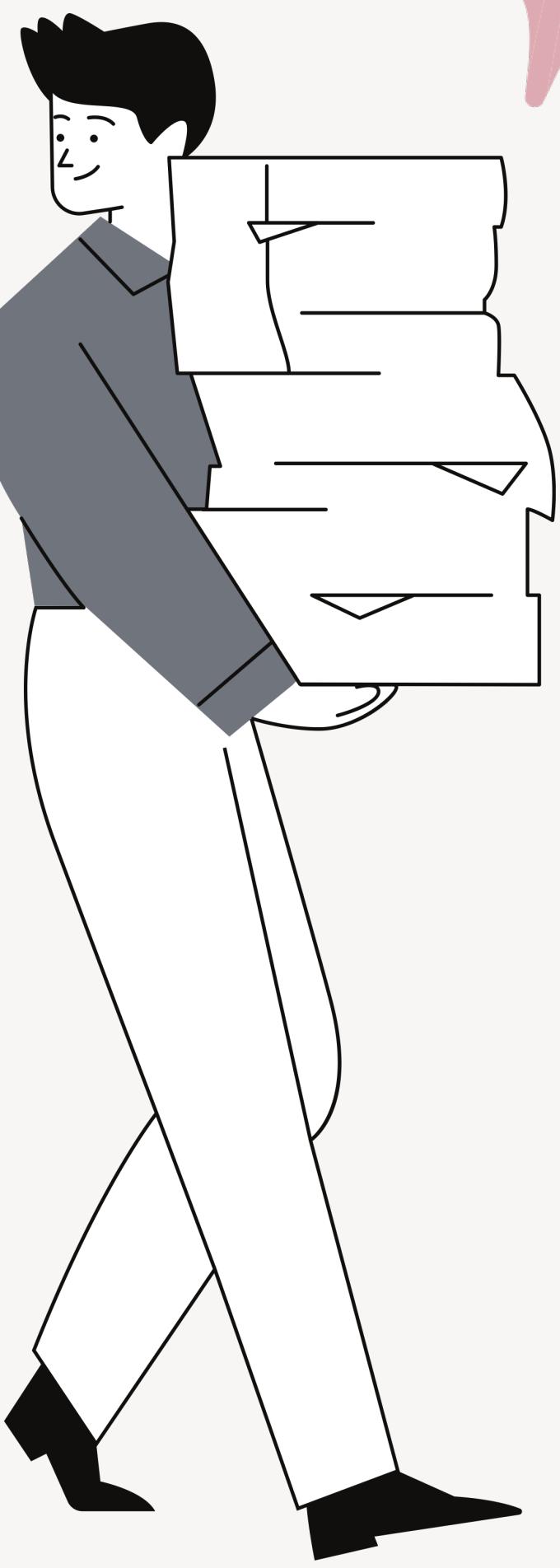


Section of Pathology and Tumour Biology, University of Leeds

# Digital Pathology



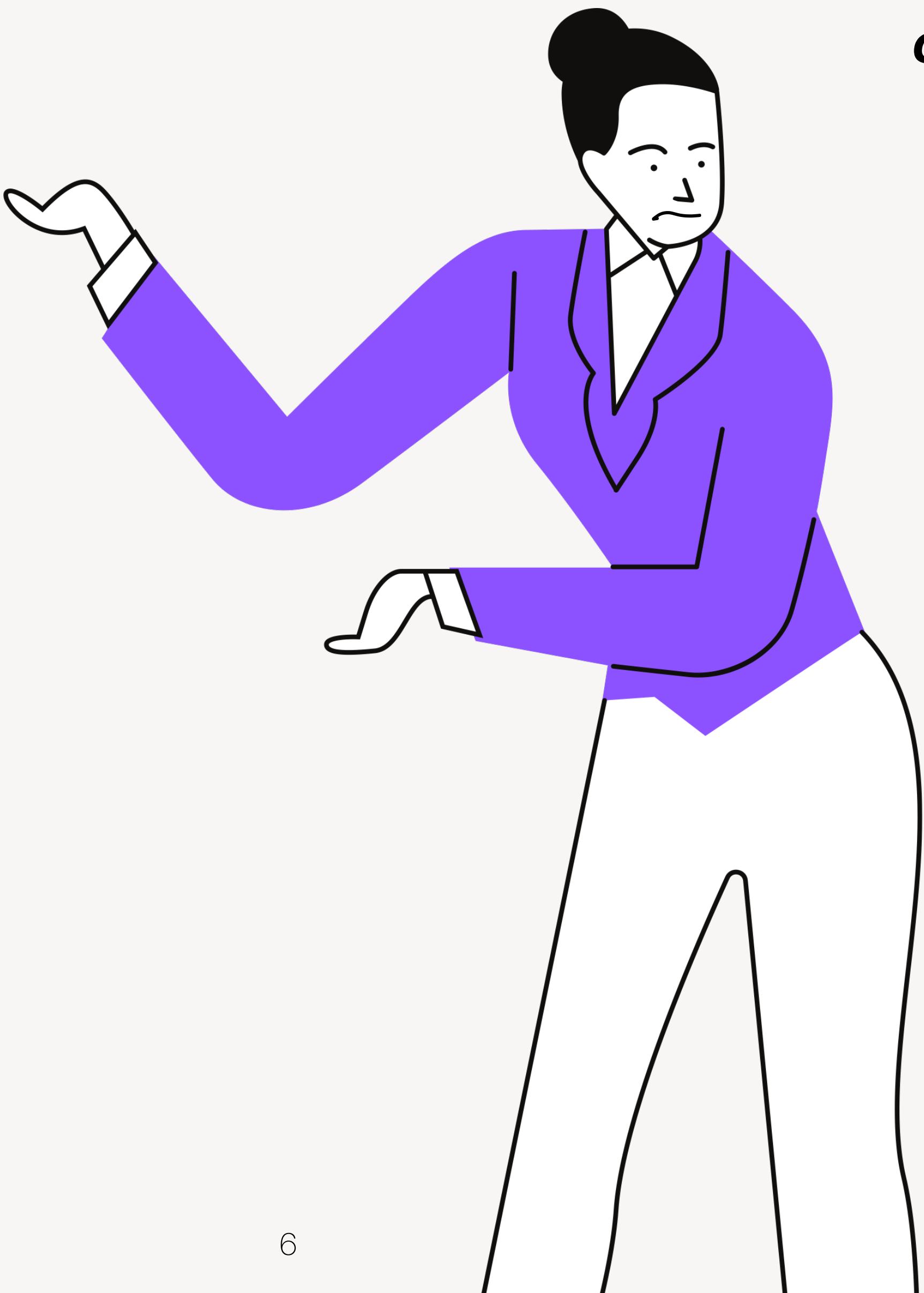
I am so excited to work  
on this new digital  
pathology project!!



Let's find some  
literature first!

day 1

**\*Reading papers  
without any mention  
of repo containing the  
source code\***



**day 3**



**Yay, finally some code!**

**day 5**

*\*opens code\**



day 5

\*ope

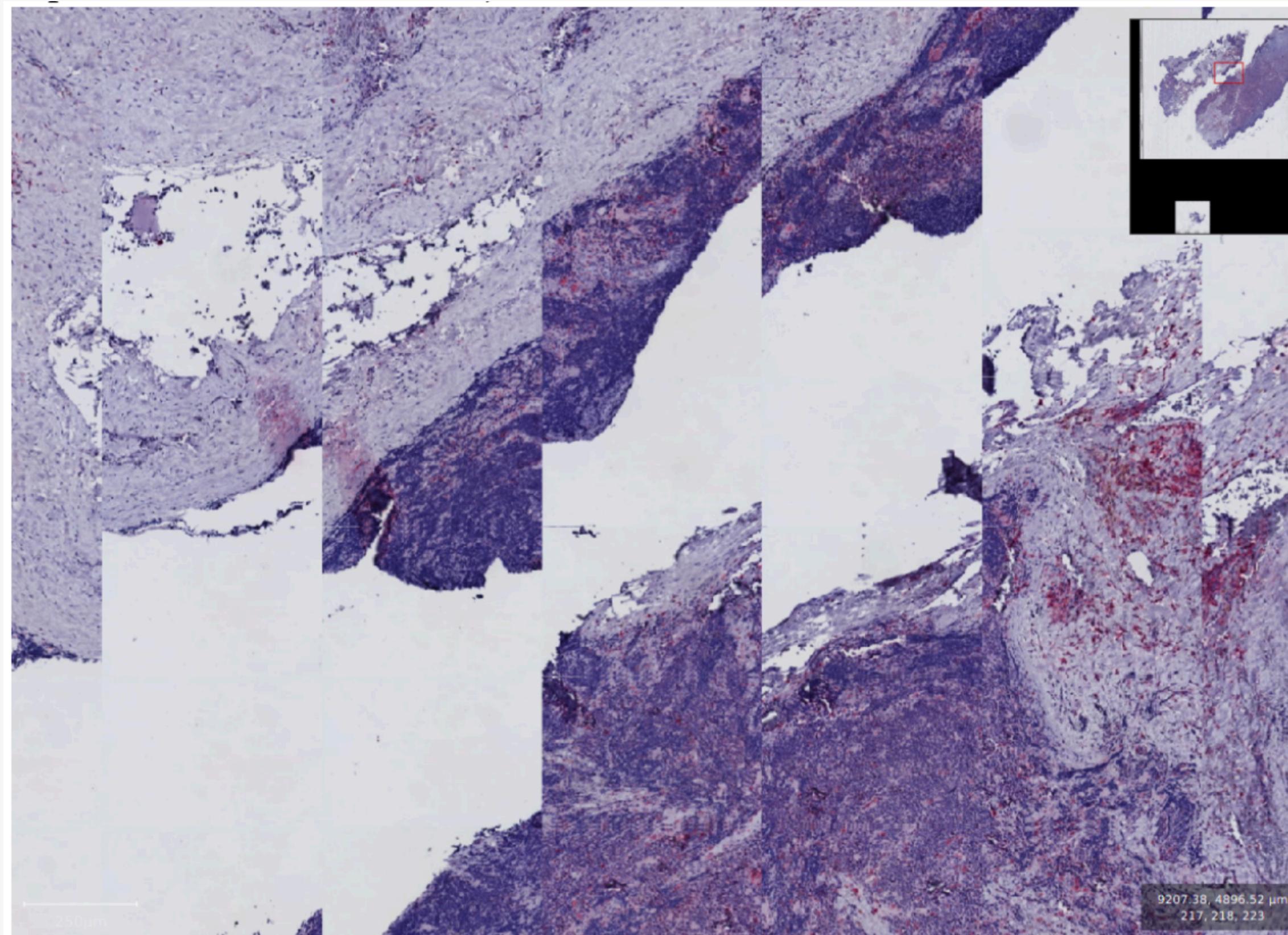
# DATASET PREPROCESSING CLASS

IF THEN  
ELSE, HARDCODED  
PATHS, INTERFACE  
TO CUSTOM  
FORMATS 1000  
LINES OF  
COPY-PASTED JUPYTER CODE



day 5

## **“Tagliatelle” effect, 2017**



Alessia Marcolini  
@viperale

...

Winning a hard battle against [#digitalPathologyTagliatelle](#) w/ [@nicole\\_bussola](#). Now let's get the real work done!  
[#DeepLearning](#) for [#Neuroblastoma](#) TILs.  
[@Pathomation1](#)

Thanks Francesco Ciompi [@furlanello](#) [@mpbalab](#)  
[@FBKcom](#) [@CIBIO\\_UniTrento](#)

[Traduci il Tweet](#)



4:57 PM · 6 dic 2019 · Twitter Web App

[Visualizza interazioni Tweet](#)

4 Retweet 32 Mi piace

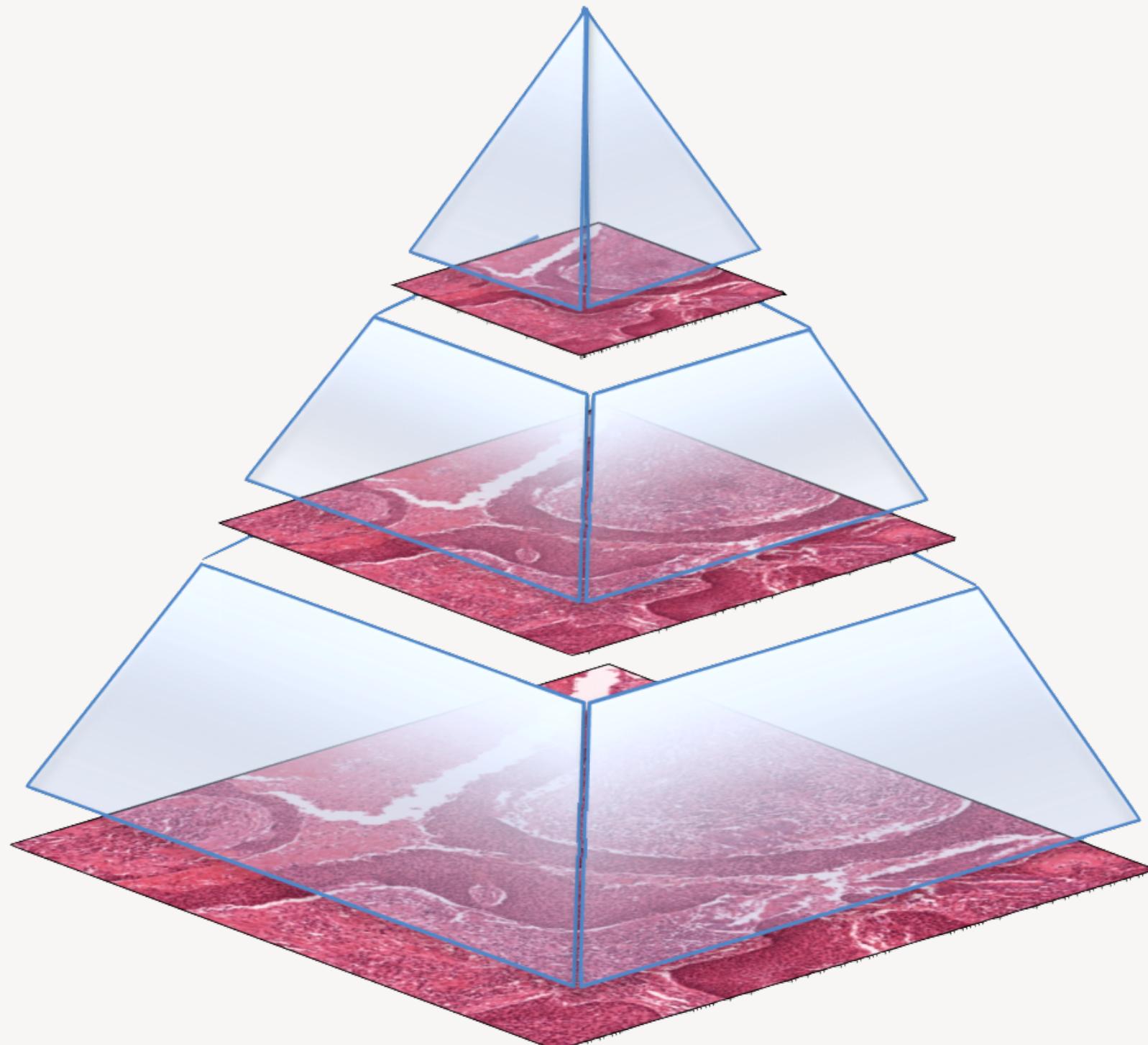
<https://twitter.com/viperale/status/1202980338693226496>

# Whole Slide Images

**Multi-resolution** image (e.g. 5 $\times$  and 20 $\times$ )  
**Pyramidal** format

**Artifacts** like **shadows, mold, pen marks**

**Ad-hoc software** for viewing and processing



Up to 90,000px  $\times$  30,000px  
**Very large in size** up to 10GB

**Scanner vendor specific file format**  
e.g. .svs, .vms, .ndpi, .tif, .bif, .scn

Image from Y. Wang et al. 2012.  
*SurfaceSlide: A multitouch digital pathology platform*  
[10.1371/journal.pone.0030783](https://doi.org/10.1371/journal.pone.0030783)

# Whole Slide Images

Artifacts like shadow, mold, pen marks

Ad-hoc software

for viewing and processing

 Trevor D. McKee PhD   
@tdmckee

Could slide scanner manufacturers just stop it with the proprietary closed file formats? I got told that a DICOM converter would "cost extra, talk to sales" - come on. Just let us make ome.tiff files please. I'm a half page of python code into making a converter, but its annoying

[Traduci il Tweet](#)

12:31 PM · 5 nov 2021 da Toronto, Ontario · Twitter for Android

---

1 Retweet 1 Tweet di citazione 12 Mi piace

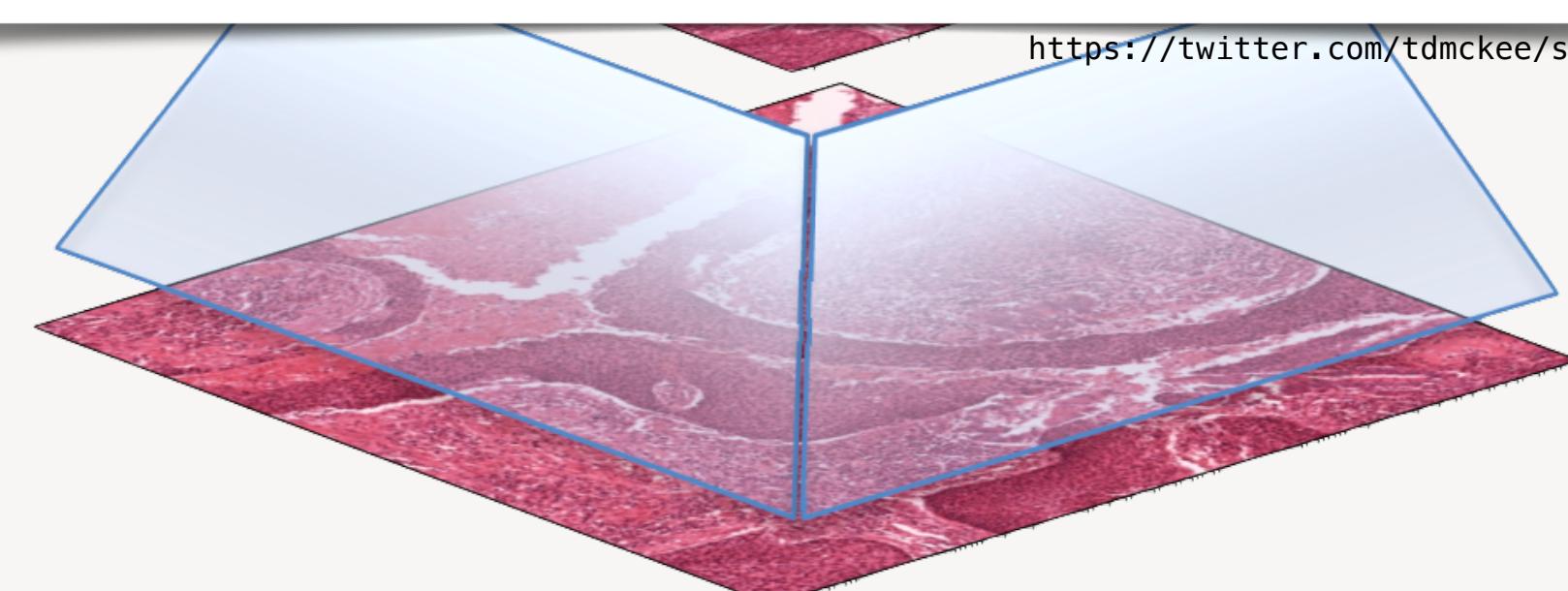


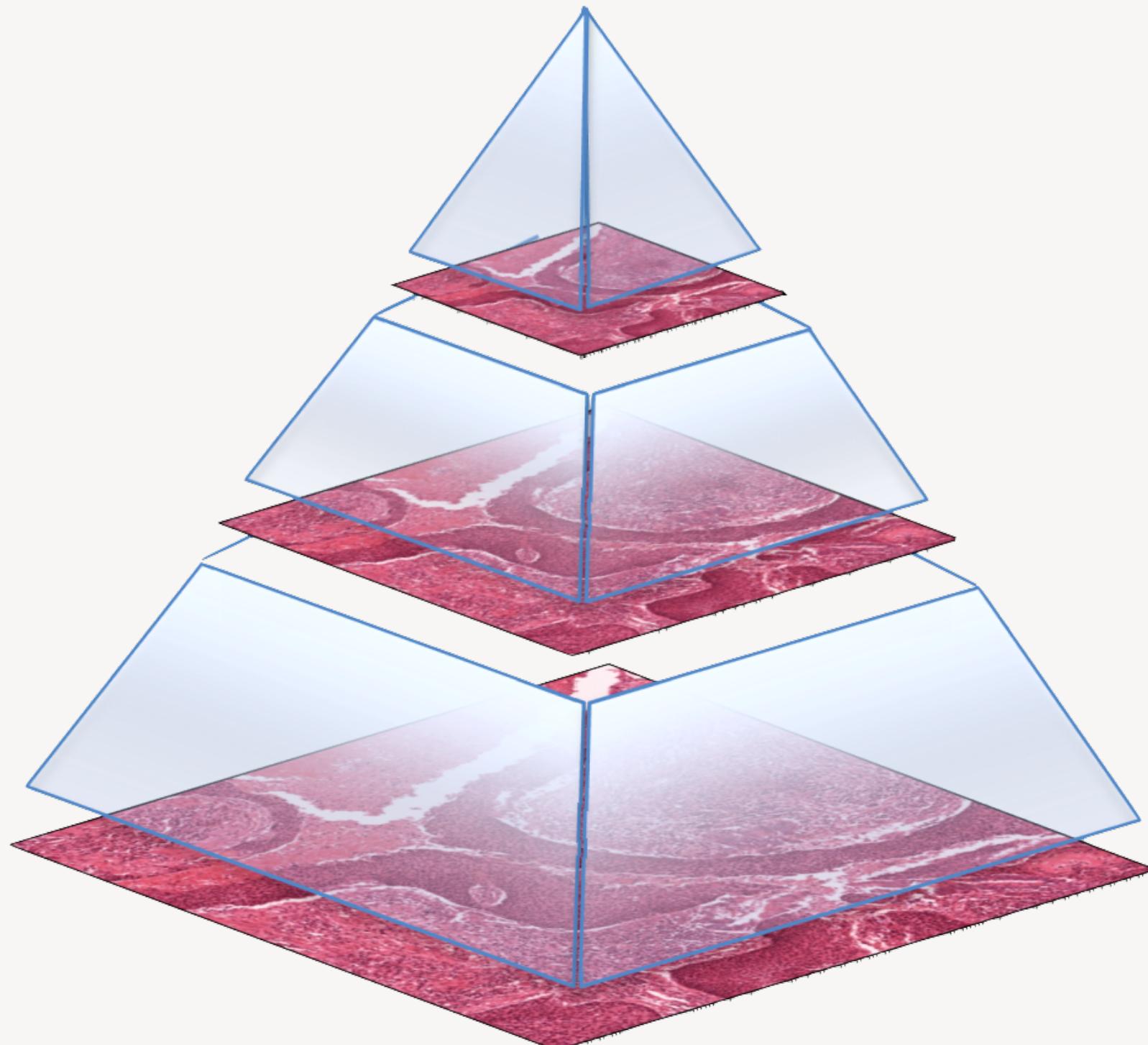
Image from Y. Wang et al. 2012.  
*SurfaceSlide: A multitouch digital pathology platform*  
[10.1371/journal.pone.0030783](https://doi.org/10.1371/journal.pone.0030783)

# Whole Slide Images

**Multi-resolution** image (e.g. 5 $\times$  and 20 $\times$ )  
**Pyramidal** format

**Artifacts** like **shadows, mold, pen marks**

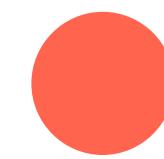
**Ad-hoc software** for viewing and processing



Up to 90,000px  $\times$  30,000px  
**Very large in size** up to 10GB

**Scanner vendor specific file format**  
e.g. .svs, .vms, .ndpi, .tif, .bif, .scn

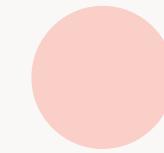
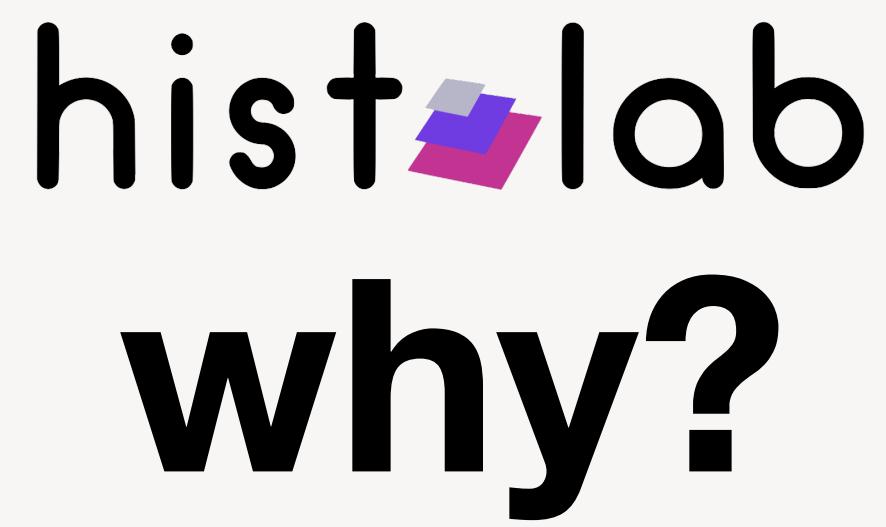
Image from Y. Wang et al. 2012.  
*SurfaceSlide: A multitouch digital pathology platform*  
[10.1371/journal.pone.0030783](https://doi.org/10.1371/journal.pone.0030783)



**Using WSIs directly as input to DL is unfeasible**

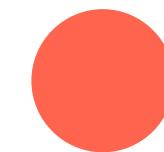
Preprocessing to create smaller subwindows ("tiles") is required

hist<sup>z</sup>lab  
why?



**Using WSIs directly as input to DL is unfeasible**

Preprocessing to create smaller subwindows ("tiles") is required



**Preprocessing steps usually poorly detailed in research papers**

Leading to results that are hard to reproduce

hist<sup>lab</sup>  
why?

# Preprocessing Leading to results

Using WSIs d

Preprocessing to c

Search

Browse State-of-the-Art Datasets Methods More ▾

Trending Research

Trending Latest Greatest

Subscribe

**Swin Transformer: Hierarchical Vision Transformer using Shifted Windows**

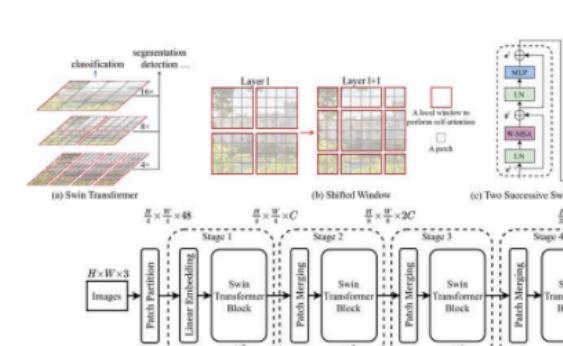
25 Mar 2021 • microsoft/Swin-Transformer • PyTorch

This paper presents a new vision Transformer, called Swin Transformer, that capably serves as a general-purpose backbone for computer vision.

Ranked #1 on Semantic Segmentation on ADE20K

IMAGE CLASSIFICATION INSTANCE SEGMENTATION

REAL-TIME OBJECT DETECTION SEMANTIC SEGMENTATION



**Is Space-Time Attention All You Need for Video Understanding?**

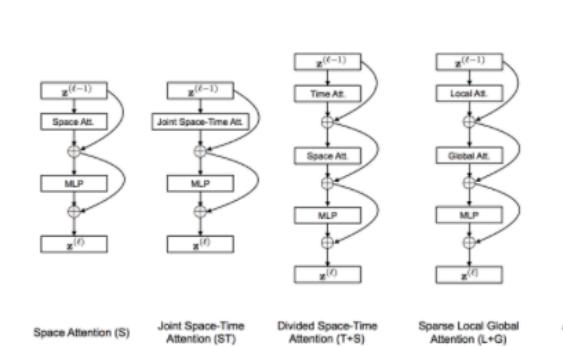
9 Feb 2021 • facebookresearch/TimeSformer • PyTorch

We present a convolution-free approach to video classification built exclusively on self-attention over space and time.

Ranked #1 on Action Recognition on Diving-48

ACTION CLASSIFICATION ACTION RECOGNITION

VIDEO QUESTION ANSWERING VIDEO UNDERSTANDING



<https://paperswithcode.com/>

12

## Papers Without Code - submission

Papers Without Code: where unreproducible papers come to live.

The goal of this is to save the time and effort of researchers who try to reproduce the results of a paper that is unreproducible. It could either be due to the paper not having enough details or the method straight up not working. In either case, authors will be given the opportunity to respond. The hope is this saves people time and disincentivizes unreproducible papers.

1. First authors of the paper will be informed and given a chance to respond.
2. Submissions with multiple votes and/or a link to a reproduction will be given priority.
3. Every submission will be reviewed to prevent spam. If it is a genuine submission, expect it to be approved within 24 hours.

Note: In order to protect the authors reputation, we will take spamming very seriously.

View all submitted papers:

<https://airtable.com/shrWz8OF3uMZ8G4cY>

Contact: [papersburned@gmail.com](mailto:papersburned@gmail.com)

Results: [papers.paperswithoutcode.com](http://papers.paperswithoutcode.com)

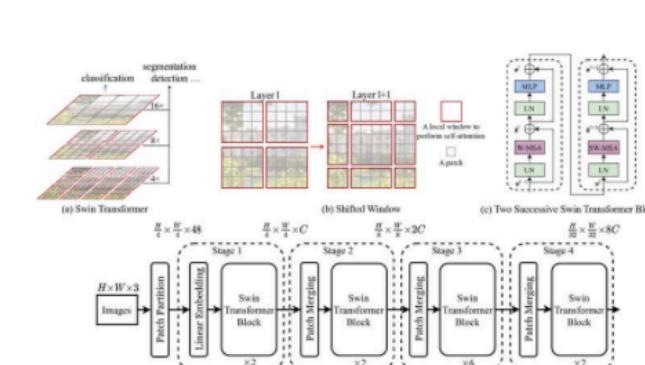
Reddit community: <http://reddit.com/r/paperswithoutcode>

<https://www.paperswithoutcode.com/>

## Trending Research

Trending Latest Greatest

 [Subscribe](#)



### Swin Transformer: Hierarchical Vision Transformer using Shifted Windows

25 Mar 2021 • microsoft/Swin-Transformer • PyTorch

This paper presents a new vision Transformer, called Swin Transformer, that capably serves as a general-purpose backbone for computer vision.

 2,604

1.73 stars / hour

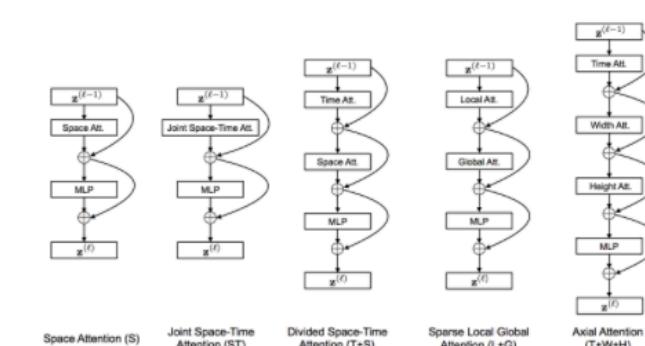
 [Paper](#)

 [Code](#)

 Ranked #1 on [Semantic Segmentation on ADE20K](#)

IMAGE CLASSIFICATION INSTANCE SEGMENTATION

REAL-TIME OBJECT DETECTION SEMANTIC SEGMENTATION



### Is Space-Time Attention All You Need for Video Understanding?

9 Feb 2021 • facebookresearch/TimeSformer • PyTorch

We present a convolution-free approach to video classification built exclusively on self-attention over space and time.

 243

1.72 stars / hour

 [Paper](#)

 [Code](#)

 Ranked #1 on [Action Recognition on Diving-48](#)

ACTION CLASSIFICATION ACTION RECOGNITION

VIDEO QUESTION ANSWERING VIDEO UNDERSTANDING

<https://paperswithcode.com/>

# hist<sup>lab</sup> why?



**Using WSIs directly as input to DL is unfeasible**

Preprocessing to create smaller subwindows ("tiles") is required



**Preprocessing steps usually poorly detailed in research papers**

Leading to results that are hard to reproduce



**Need for a reference high quality preprocessing software**

To enable faster prototyping and faster experimentation

## Using WSIs directly as input to DL is unfeasible

Alex Kyllo  
@alexkyllo

Software engineering for data science is such a glaring, unsolved problem. I've worked for years at the intersection of the two and am still amazed by how difficult it is to write good, maintainable code for data science and how no one really even knows what that means yet.

[Traduci il Tweet](#)

8:47 PM · 6 nov 2021 · Twitter Web App

178 Retweet 33 Tweet di citazione 1.728 Mi piace

Alex Kyllo  
@alexkyllo

I've spent a lot of effort learning software engineering skills & best practices and yet still, every serious DS project I start eventually becomes a big ball of mud and I'm constantly disappointed by own inability to avoid this even when I realize it's happening in real time.

[Traduci il Tweet](#)

8:51 PM · 6 nov 2021 · Twitter Web App

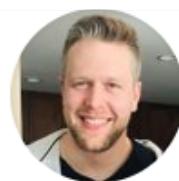
3 Retweet 3 Tweet di citazione 192 Mi piace

<https://twitter.com/alexkyllo/status/1457072262520004632>

## Need for a reference high quality preprocessing software

To enable faster prototyping and faster experimentation

## Using WSIs directly as input to DL is unfeasible



Alex Kyllo  
@alexkyllo

Software engineering for data science is such a glaring, unsolved problem. I've worked for years at the intersection of the two and am still amazed by how difficult it is to write good, maintainable code for data science and how no one really even knows what that means yet.

[Traduci il Tweet](#)

8:47 PM · 6 nov 2021 · Twitter Web App

178 Retweet 33 Tweet di citazione 1.728



Alex Kyllo  
@alexkyllo

I've spent a lot of effort learning software engineering skills & best practices and yet still, every serious DS project I start eventually becomes a big ball of mud and I'm constantly disappointed by own inability to avoid this even when I realize it's happening in real time.



Pete Bankhead  
@petebankhead

I find it a bit sad that in academia the proof-of-concept is often the glamorous (/fundable/publishable) bit, since I've always found the next steps \*far\* harder.

#ResearchSoftwareEngineering is tricky.

[Traduci il Tweet](#)

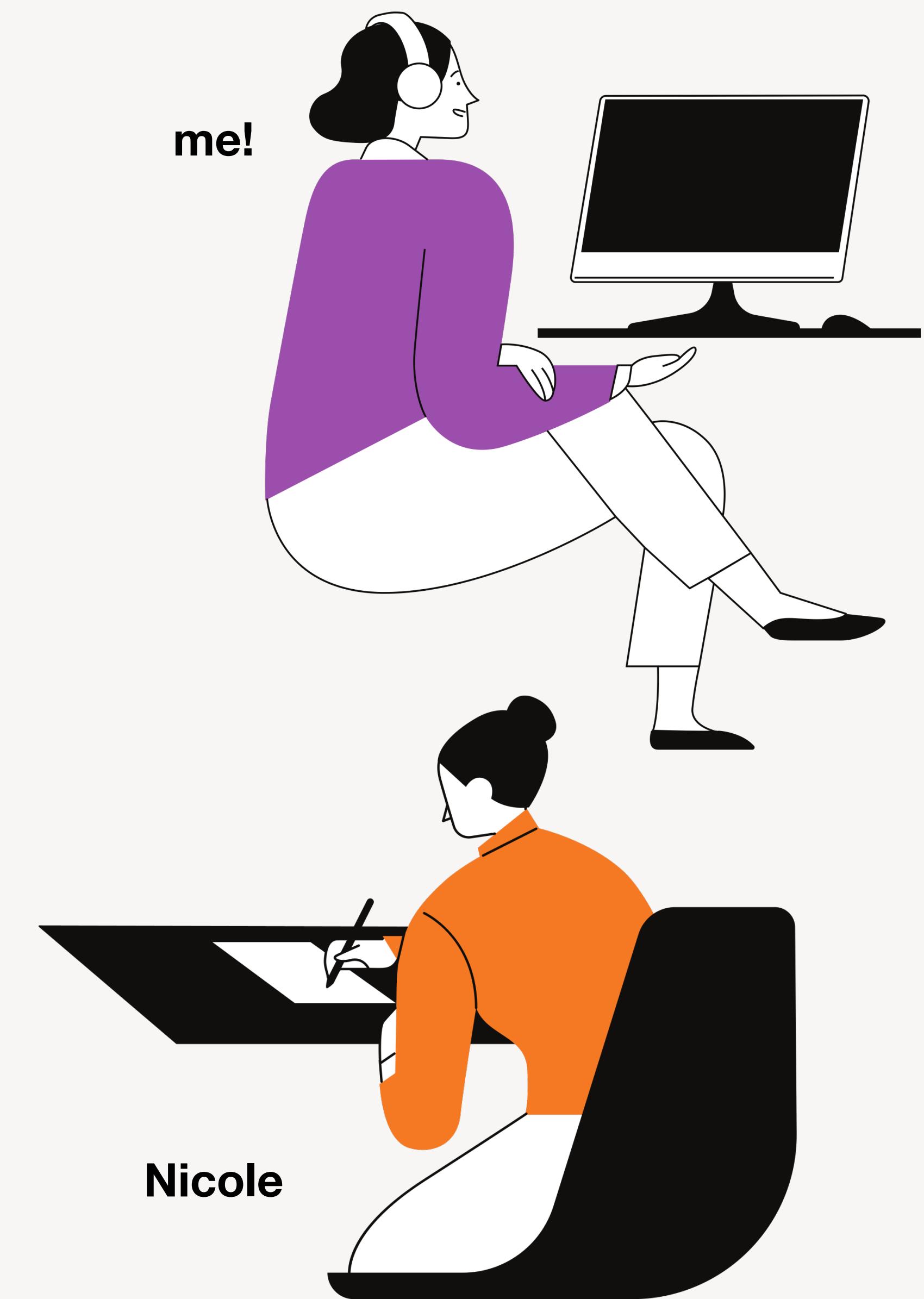
11:24 AM · 23 giu 2021 · Twitter Web App

1 Retweet 15 Mi piace

cessing software

entation

<https://twitter.com/alexkyllo/status/1457072262520004632>



**Nicole**

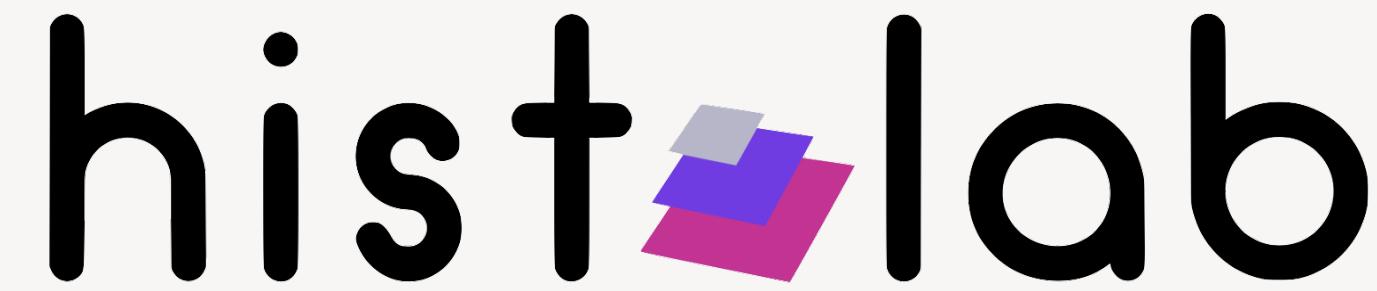


**Ernesto**

hist<sup>+</sup> lab

team

day 60

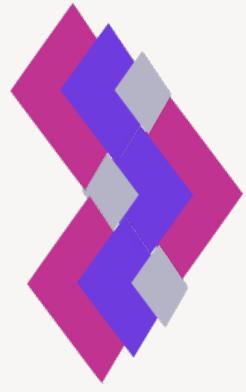


new open source Python package for  
**reproducible Whole Slide Images preprocessing**

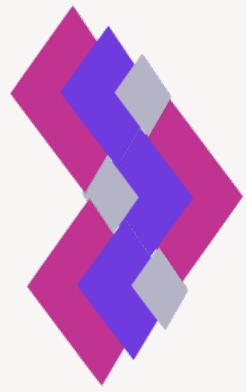


aimed at an easy  
**integration with a Deep Learning pipeline**

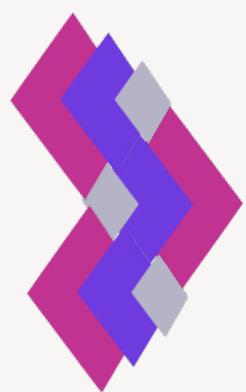
# hist<sup>+</sup>lab approach



unifying **community-validated** procedures  
for **slide preprocessing** and **tiles extraction**



introducing **best practices from software engineering**: automated testing, code versioning and code reviews, Continuous Integration



on top of **state-of-the-art** and **well-known libraries**,  
e.g. OpenSlide, NumPy and scikit-image

# Histolab features

#1

**Interoperability  
between different  
formats**

up to 9 supported  
formats from the  
major scanner  
vendors

# Histolab features

#1

**Interoperability  
between different  
formats**

up to 9 supported  
formats from the  
major scanner  
vendors

#2

**Automatic tissue  
detection and  
segmentation**

by using a fixed  
sequence of image  
filters

# Histolab features

#1

Interoperability  
between different  
formats

up to 9 supported  
formats from the  
major scanner  
vendors

#2

Automatic tissue  
detection and  
segmentation

by using a fixed  
sequence of image  
filters

#3

**Automatic  
informative tiles  
retrieval**

cropped regions  
from tissue areas  
found in #2

# Histolab features

#1

**Interoperability  
between different  
formats**

up to 9 supported  
formats from the  
major scanner  
vendors

#2

**Automatic tissue  
detection and  
segmentation**

by using a fixed  
sequence of image  
filters

#3

**Automatic  
informative tiles  
retrieval**

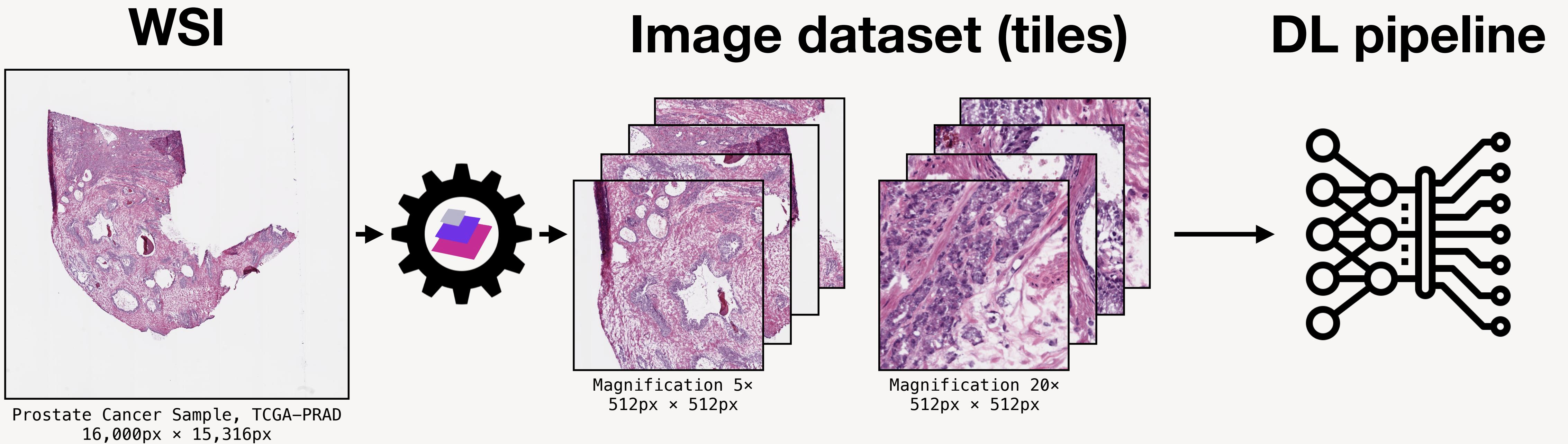
cropped regions  
from tissue areas  
found in #2

#4

**Easy access to  
sample data  
from TCGA and  
OpenSlide**

save to the system  
cache and import  
them

# Histolab in action



# Tiles extraction #3

in less than 10 lines of code

```
>>> from histolab.data import breast_tissue  
>>> _, path = breast_tissue()
```

1. download breast  
tissue sample from  
TCGA

```
>>> from histolab.slide import Slide  
>>> slide = Slide(path, "path/to/processed")
```

2. create a **Slide** object

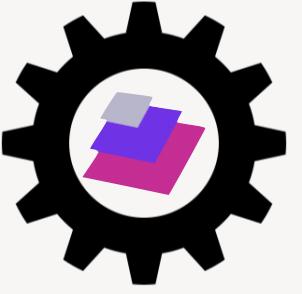
```
>>> from histolab.tiler import RandomTiler  
>>> random_tiles_extractor = RandomTiler(  
    tile_size=(512, 512),  
    n_tiles=10,  
    level=2,  
    seed=42,  
    check_tissue=True,  
)
```

3. create a **Tiler**

```
>>> random_tiles_extractor.extract(slide)
```

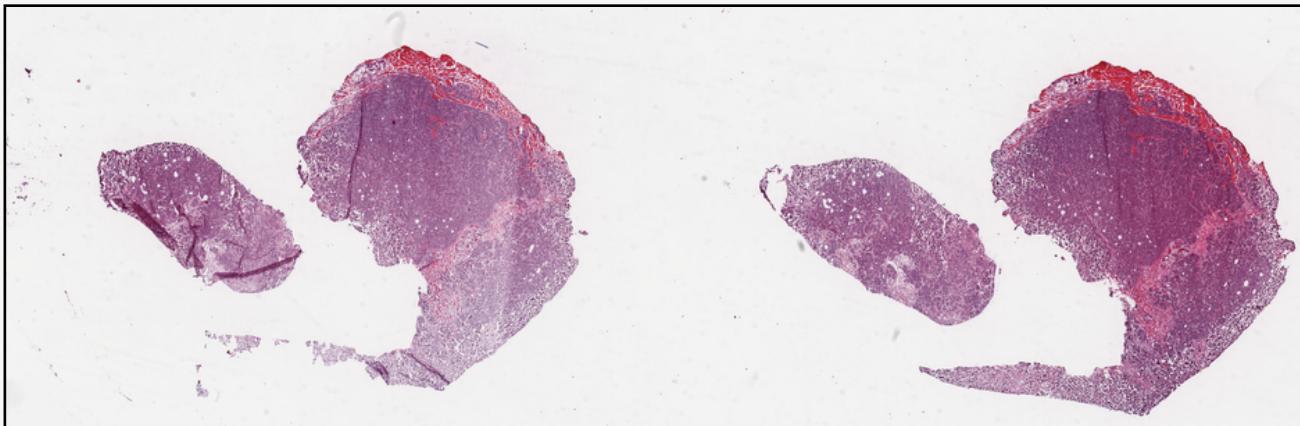
4. extract!

# Tissue detection and tiles extraction



RandomTiler

Original WSI

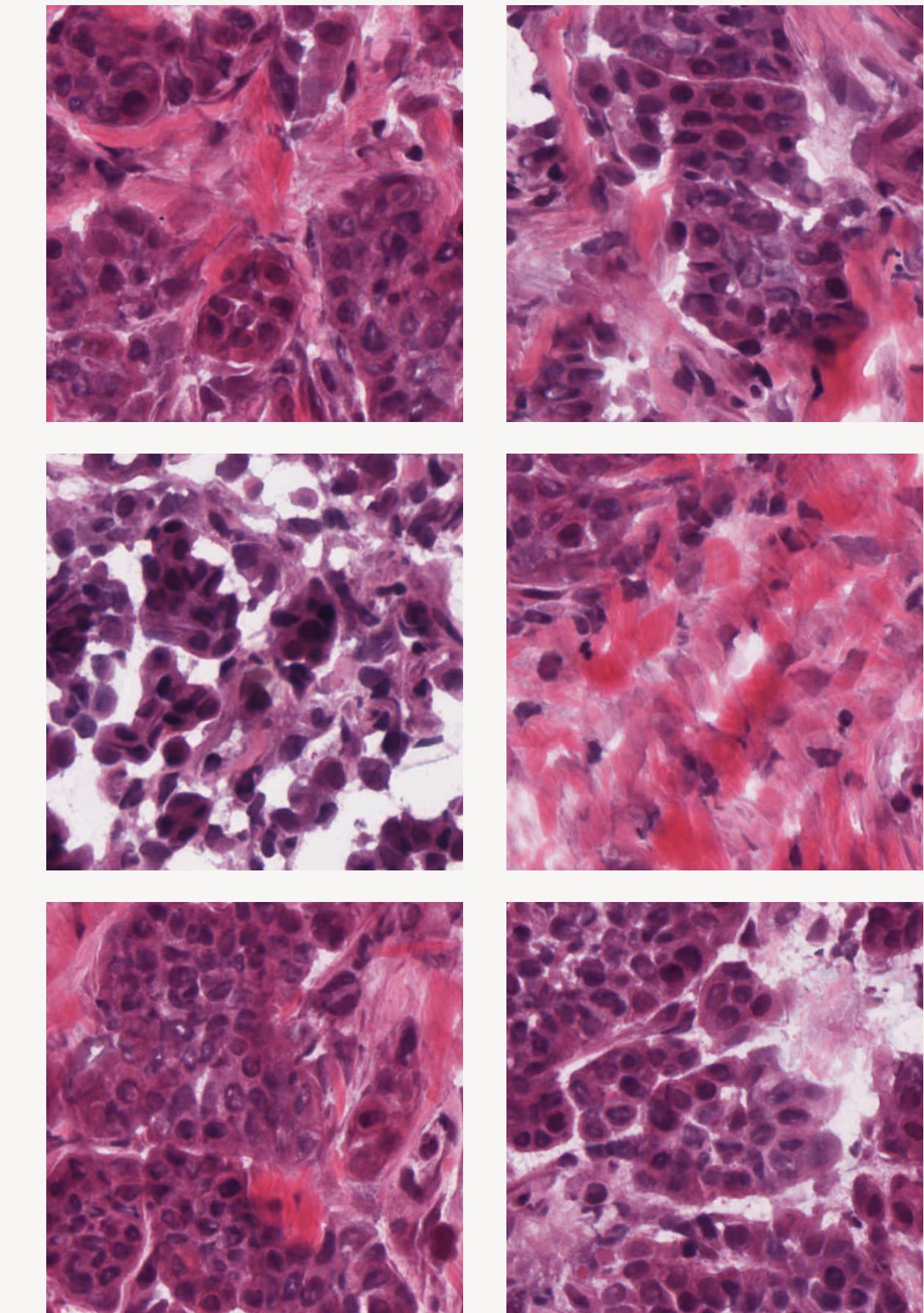


Breast Cancer Sample, TCGA-BRCA  
96,972px × 30,682px

Tissue  
Detection

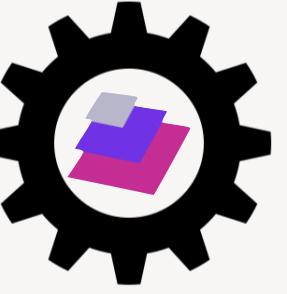


Random Tiles



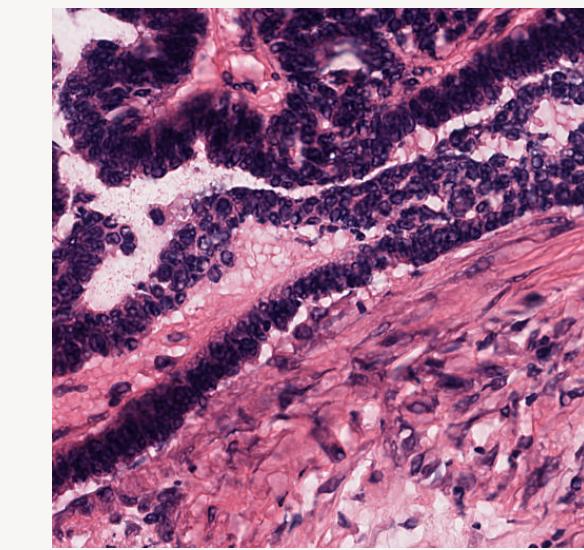
Magnification 20×  
512px × 512px

# Tissue detection and tiles extraction

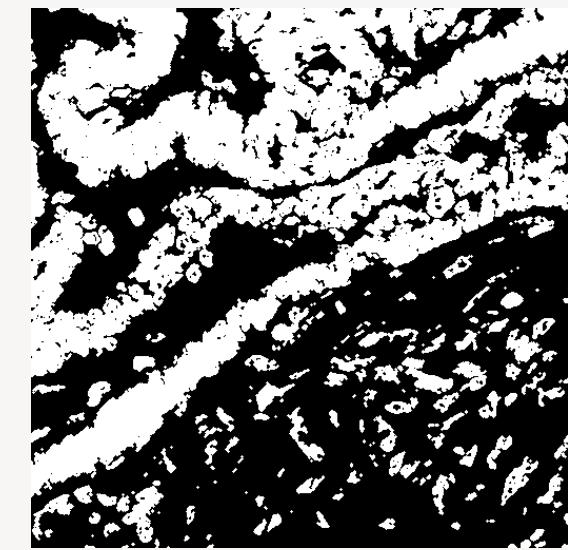


ScoreTiler

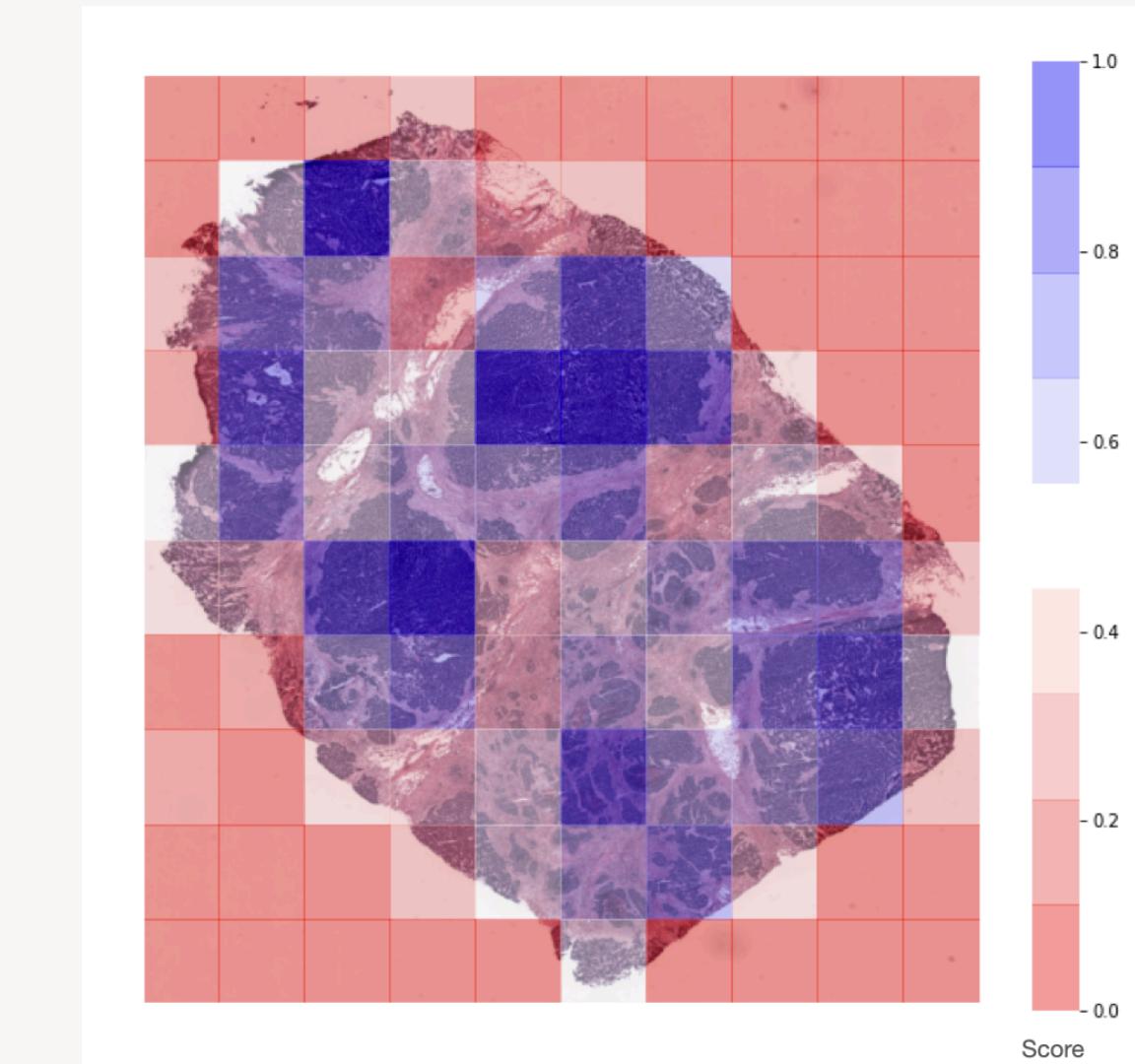
```
>>> from histolab.scorer import NucleiScorer  
>>> scorer = NucleiScorer()  
  
>>> from histolab.tiler import ScoreTiler  
>>> scored_tiles_extractor = ScoreTiler(  
        scorer,  
        tile_size=(512, 512),  
        n_tiles=10,  
        level=2,  
        seed=42,  
        check_tissue=True,  
)  
>>> scored_tiles_extractor.extract(slide)
```



512px × 512px tile



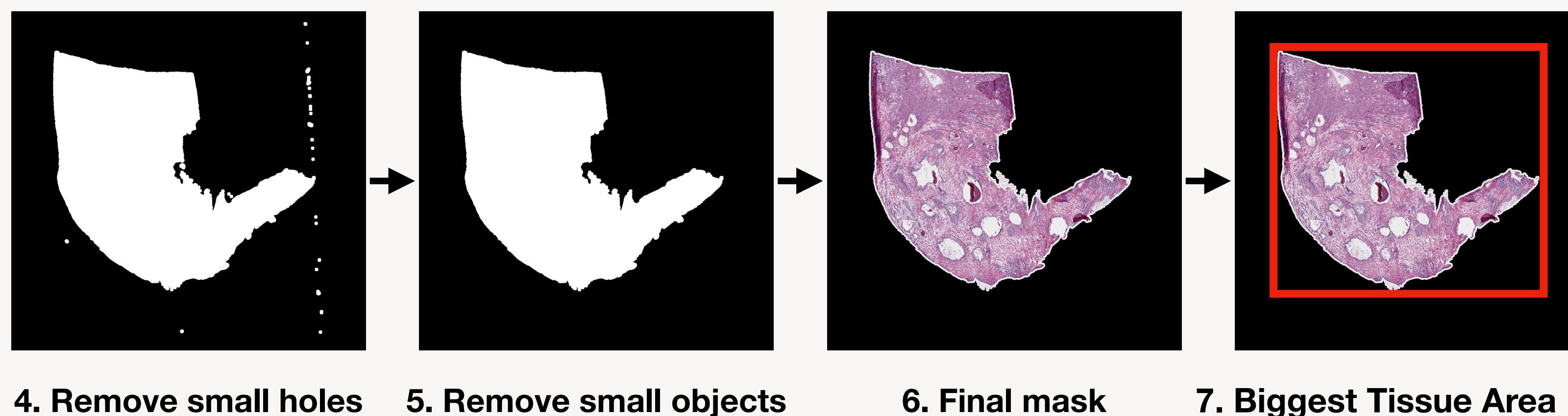
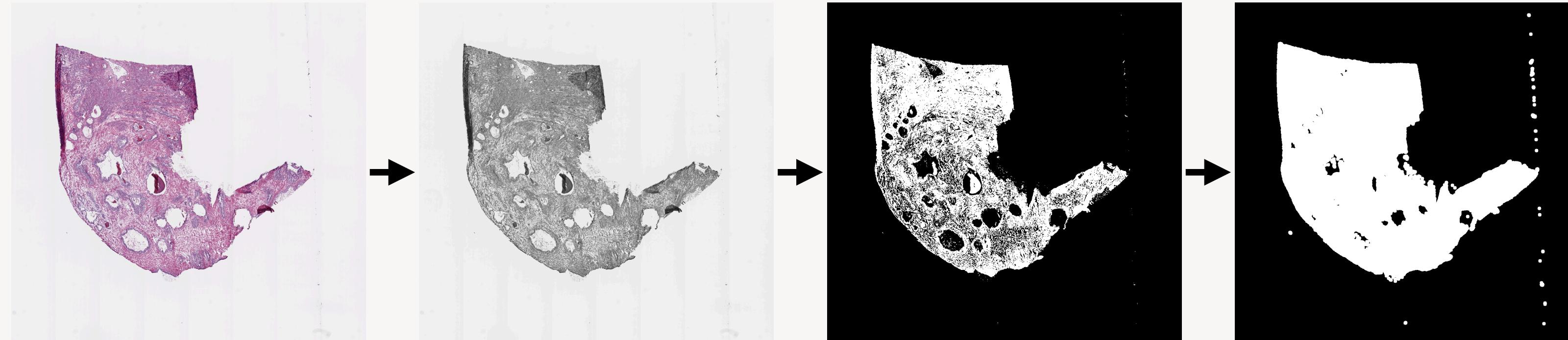
Nuclei Mask



Representation of the score assigned to each extracted tile by the NucleiScorer.

Ovarian Cancer Sample, TCGA-OV  
30,001px × 33,987px

# Tissue detection #2

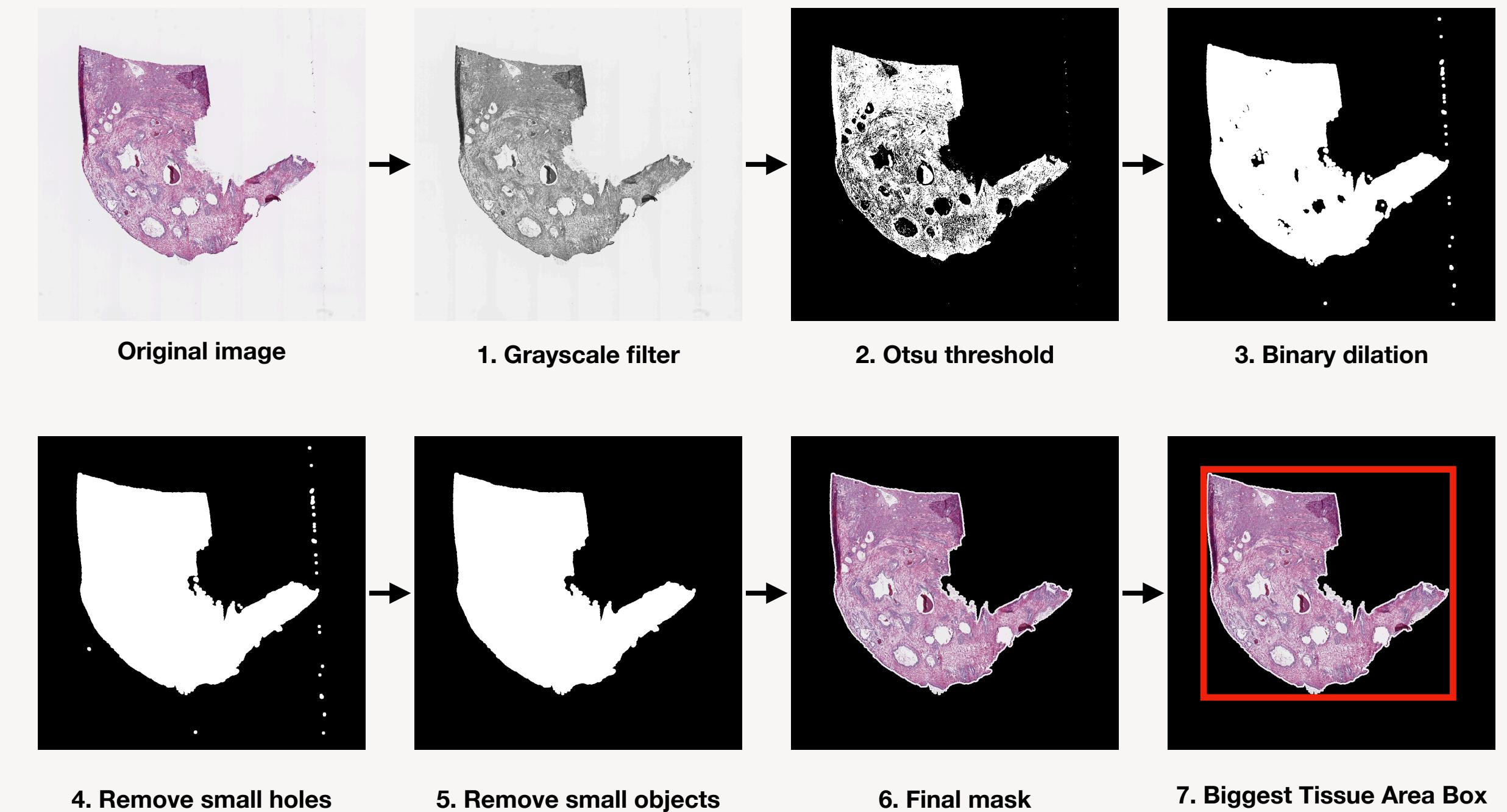


by using this fixed sequence of image filters

Nobuyuki Otsu 1979.  
A threshold selection method from gray-level histograms  
10.1109/TSMC.1979.4310076

# Tissue detection #2

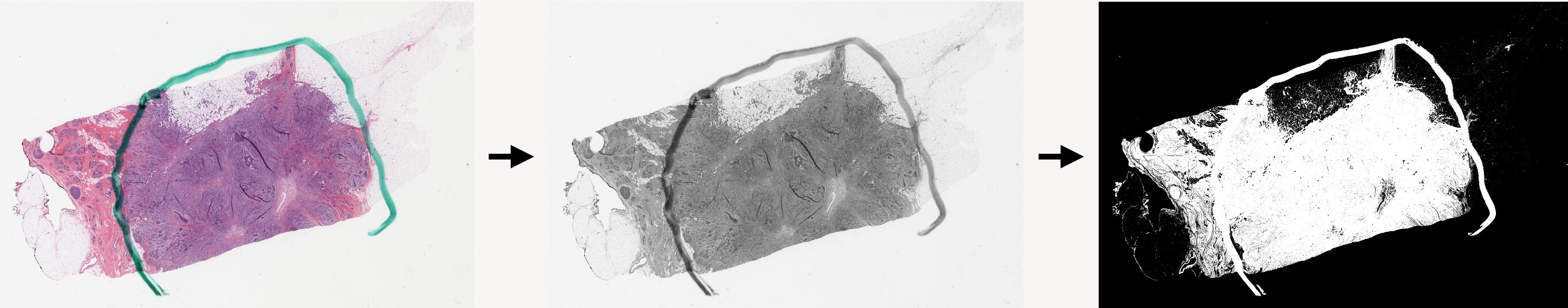
```
>>> from histolab.filters.image_filters import Compose, OtsuThreshold, RgbToGrayscale  
>>> from histolab.filters.morphological_filters import (  
    BinaryDilation,  
    RemoveSmallHoles,  
    RemoveSmallObjects,  
)  
  
>>> filters = Compose(  
    [  
        RgbToGrayscale(),  
        OtsuThreshold(),  
        BinaryDilation(),  
        RemoveSmallHoles(),  
        RemoveSmallObjects(),  
    ]  
)  
  
>>> filters(image)
```



Nobuyuki Otsu 1979.  
A threshold selection method from gray-level histograms  
[10.1109/TSMC.1979.4310076](https://doi.org/10.1109/TSMC.1979.4310076)

# Remove artifacts

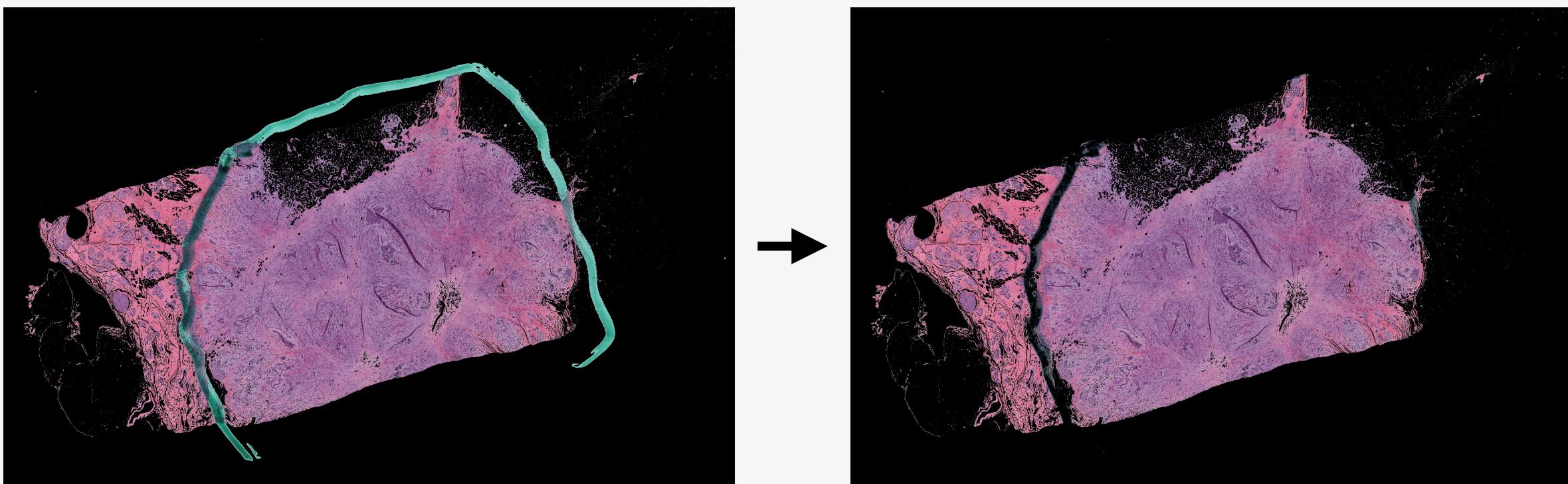
## pen markers



Original image

1. Grayscale filter

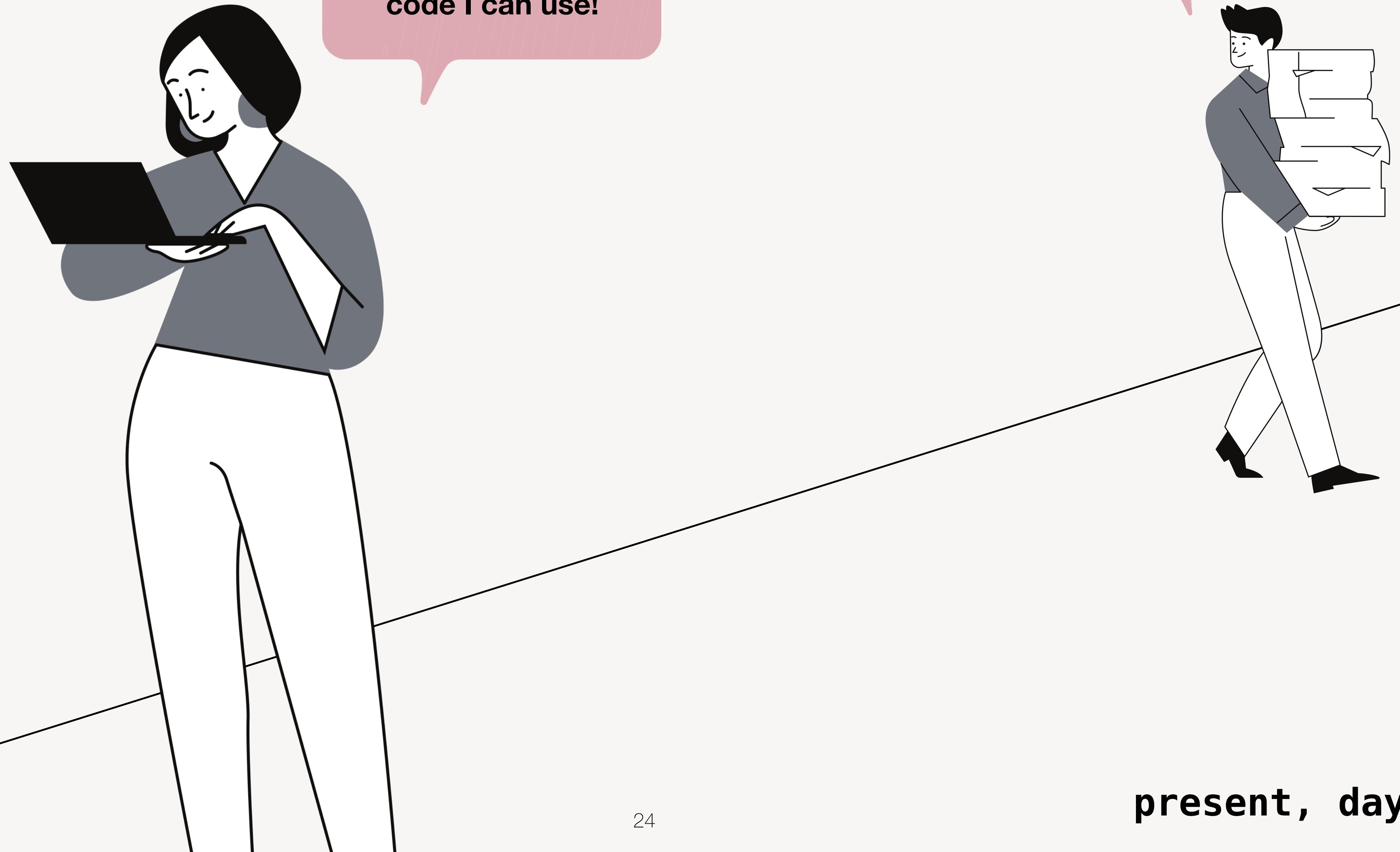
2. Otsu threshold



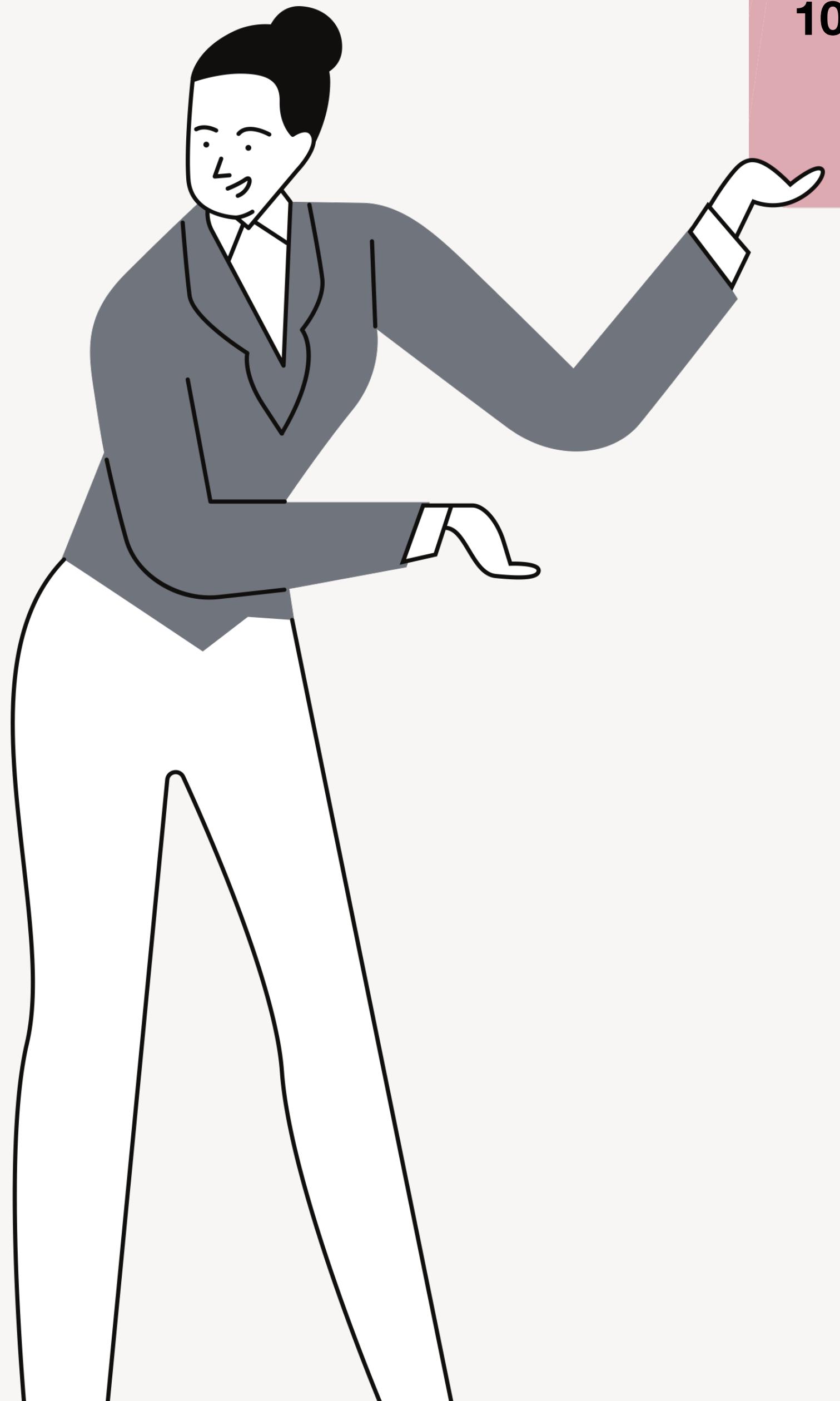
3. Apply Mask

4. Green Pen Filter

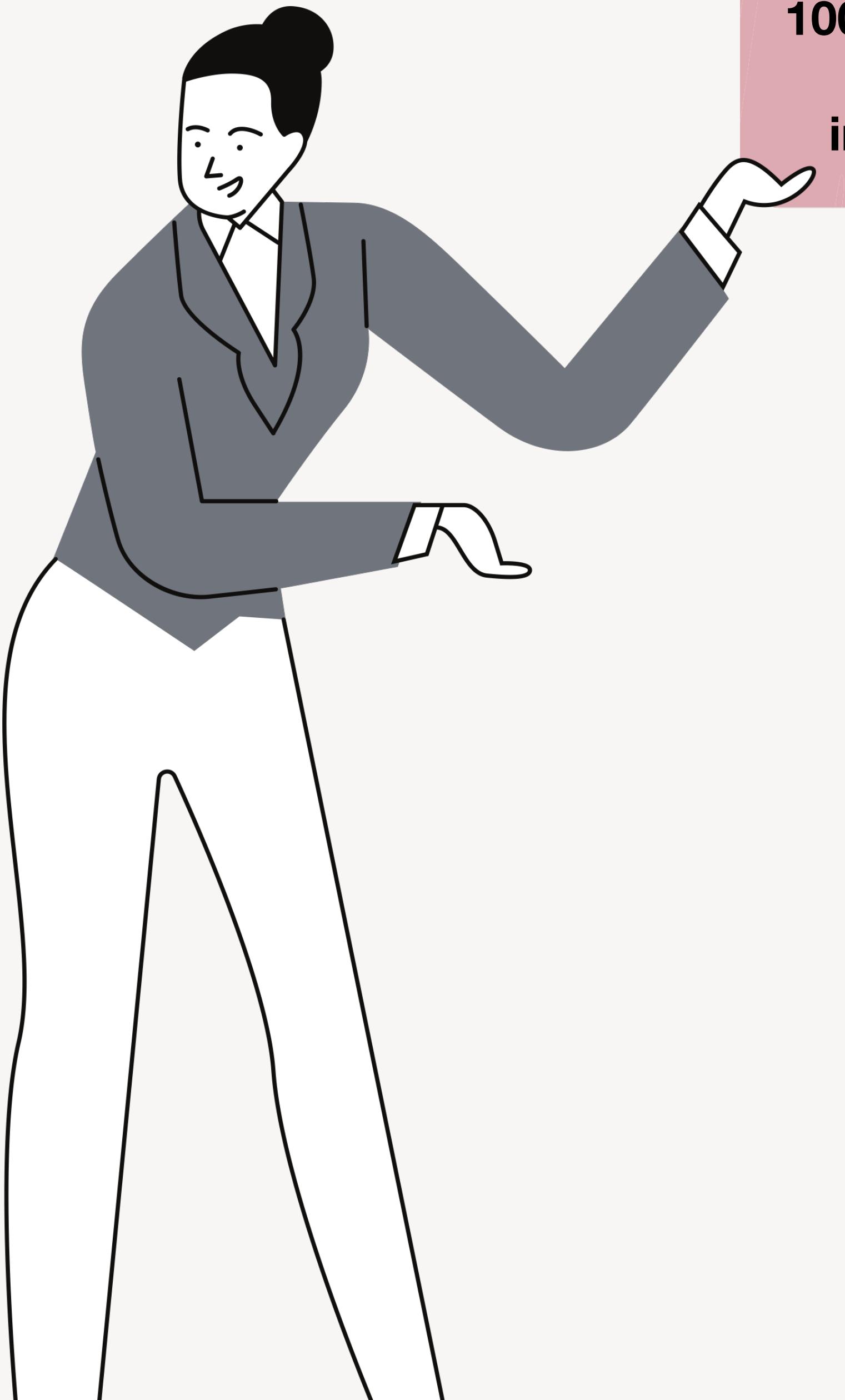
Pen Filters implementation inspired by  
<https://github.com/CODAIT/deep-histopath>



**present, day 500**



**Look at this!**  
**100% coverage with**  
**600 unit and**  
**integration tests**



**Look at this!**  
**100% coverage with**  
**600 unit and**  
**integration tests**

```

class Describe_Slide:
    @pytest.mark.parametrize(
        "slide_path, processed_path",
        [
            ("/foo/bar/myslide.svs", "/foo/bar/myslide/processed"),
            (Path("/foo/bar/myslide.svs"), Path("/foo/bar/myslide/processed")),
        ],
    )
    def it_constructs_from_args(self, request, slide_path, processed_path):
        _init_ = initializer_mock(request, Slide)

class DescribeTissueMask:
    def it_knows_its_mask(self, request, tmpdir,
                          RgbToGrayscale_, OtsuThreshold_, BinaryDilation_, RemoveSmallHoles_):
        slide, _ = base_test_slide(tmpdir, PILIMG.RGBA_COLOR_500X500_155_249_240)
        main_tissue_areas_mask_filters_ = property_mock(
            request, _SlideFiltersComposition, "tissue_mask_filters"
        )
        expected_mask = [
            [True, True],
            [False, True],
        ]
        removed_filters = []
        def test_rgb_to_hed_filter_with_rgba_image():
            img = RGBA(DIAGNOSTIC_SLIDE_THUMB)
            expected_value = load_expectation(
                "pil-images-rgba/diagnostic-slide-thumb-rgba-to-hed", type_="png"
            )
            expected_warning_regex = (
                r"Input image must be RGB. NOTE: the image will be converted to RGB before"
                r" HED conversion."
            )
            with pytest.warns(UserWarning, match=expected_warning_regex):
                hed_img = imf.rgb_to_hed(img)
            np.testing.assert_array_almost_equal(np.array(hed_img), np.array(expected_value))
            assert np.unique(np.array(ImagChops.difference(hed_img, expected_value)))[0] == 0

        def test_rgb_to_hed_raises_exception_on_gs_image():
            gs_img = GS(DIAGNOSTIC_SLIDE_THUMB_GS)

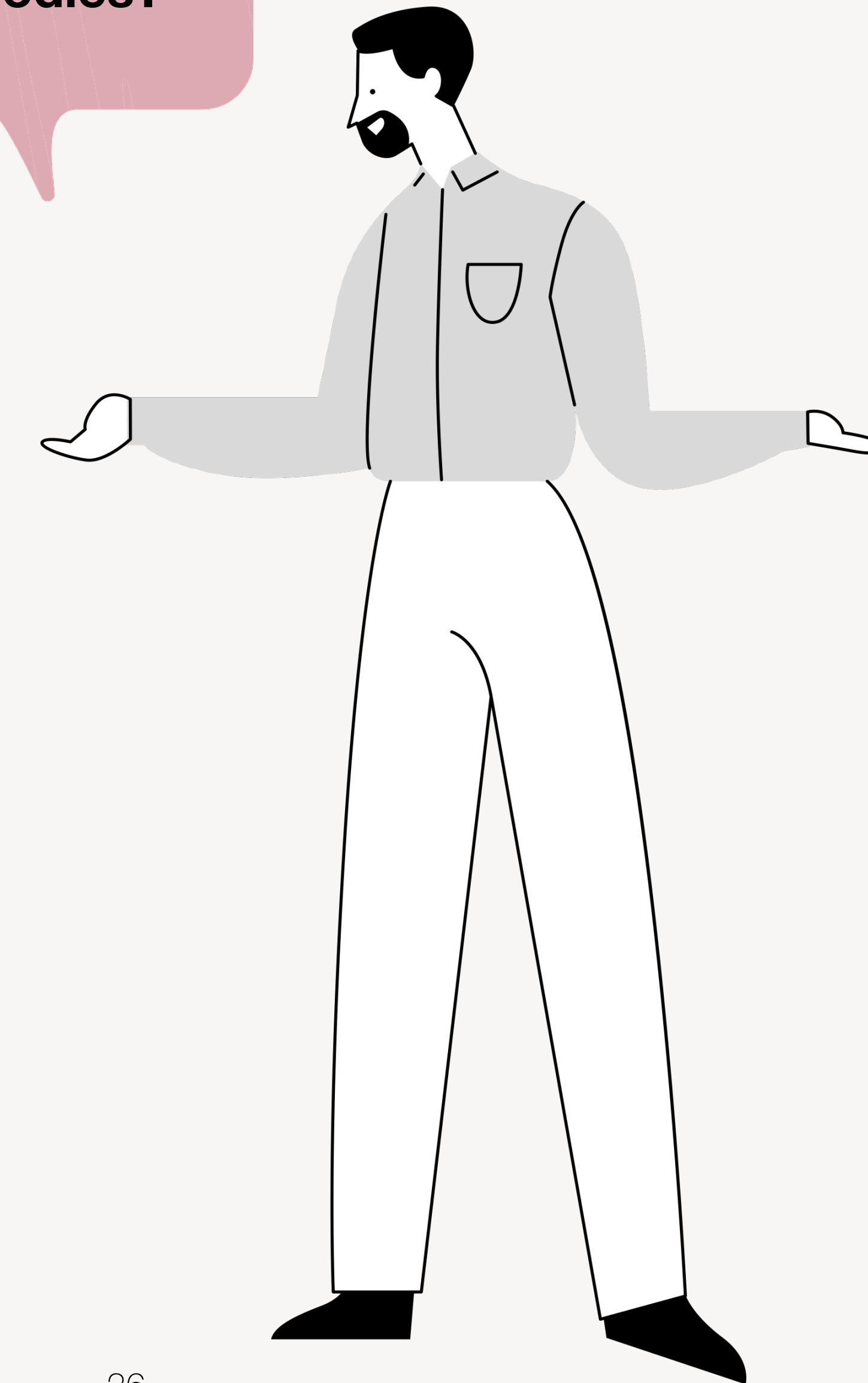
            with pytest.raises(Exception) as err:
                imf.rgb_to_hed(gs_img)

            assert isinstance(err.value, Exception)
            assert str(err.value) == "Input image must be RGB."

        def test_hematoxylin_channel_filter_with_rgb_image():
            img = RGB(TCGA_LUNG_RGB)
            expected_value = load_expectation(
                "pil-images-rgb/tcga-lung-rgb-hematoxylin-channel", type_="png"
            )
            h_channel = imf.hematoxylin_channel(img)
            np.testing.assert_array_almost_equal(np.array(h_channel), np.array(expected_value))
            assert np.unique(np.array(ImagChops.difference(h_channel, expected_value)))[0] == 0

```

**Yes, but how do I get  
all of these goodies?**



**present, day 500**

histolab / histolab Public

Sponsor Unwatch 3 Star 121 Fork 22

Code Issues 17 Pull requests 7 Discussions Actions Projects 1 Wiki Security Insights ...

master 14 branches 13 tags Go to file Add file Code

 dependabot and ernestoarbitrio Update scipy requireme... f080986 yesterday 1,031 commits

.github fix release yml 3 days ago

docs Update HED channels figure in docs 6 days ago

examples add warning in readme last month

histolab release: prepare v0.3.0 release 3 days ago

tests Move RGB->color space expectations in correct folder 4 days ago

.coveragerc TYPE\_CHECKING conditionals don't count for coverage 3 months ago

.gitignore first benchmarks 14 months ago

.pre-commit-config.yaml add bandit 2 months ago

AUTHORS rfctr: name refactoring from wsi -> slide 2 years ago

CHANGELOG.rst release: prepare v0.3.0 release 3 days ago

CONTRIBUTING.md add 3.9 reference in contributing files 25 days ago

LICENSE.txt making histolab pip-installable 2 years ago

MANIFEST.in get rid of src directory. Histolab now lives in the histolab c... last month

Makefile get rid of src directory. Histolab now lives in the histolab c... last month

README.md add note for data moudule 21 days ago

bandit.yaml address alessia's comments 2 months ago

About

Library for Digital Pathology Image Processing

histolab.readthedocs.io

python data-science bioinformatics  
research biology hacktoberfest  
digital-pathology science-research  
pathology wsi digital-pathology-data  
hacktoberfest2021

Readme

Apache-2.0 License

Releases 10

Version 0.3.0 Latest 3 days ago  
+ 9 releases

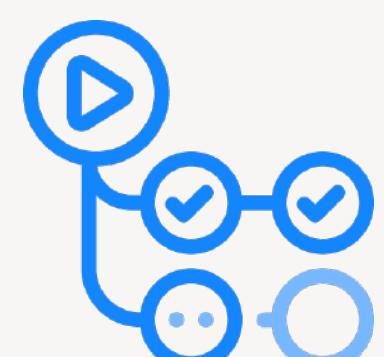
Sponsor this project

ernestoarbitrio pamaron

Sponsor

# GitHub

github.com/histolab/histolab



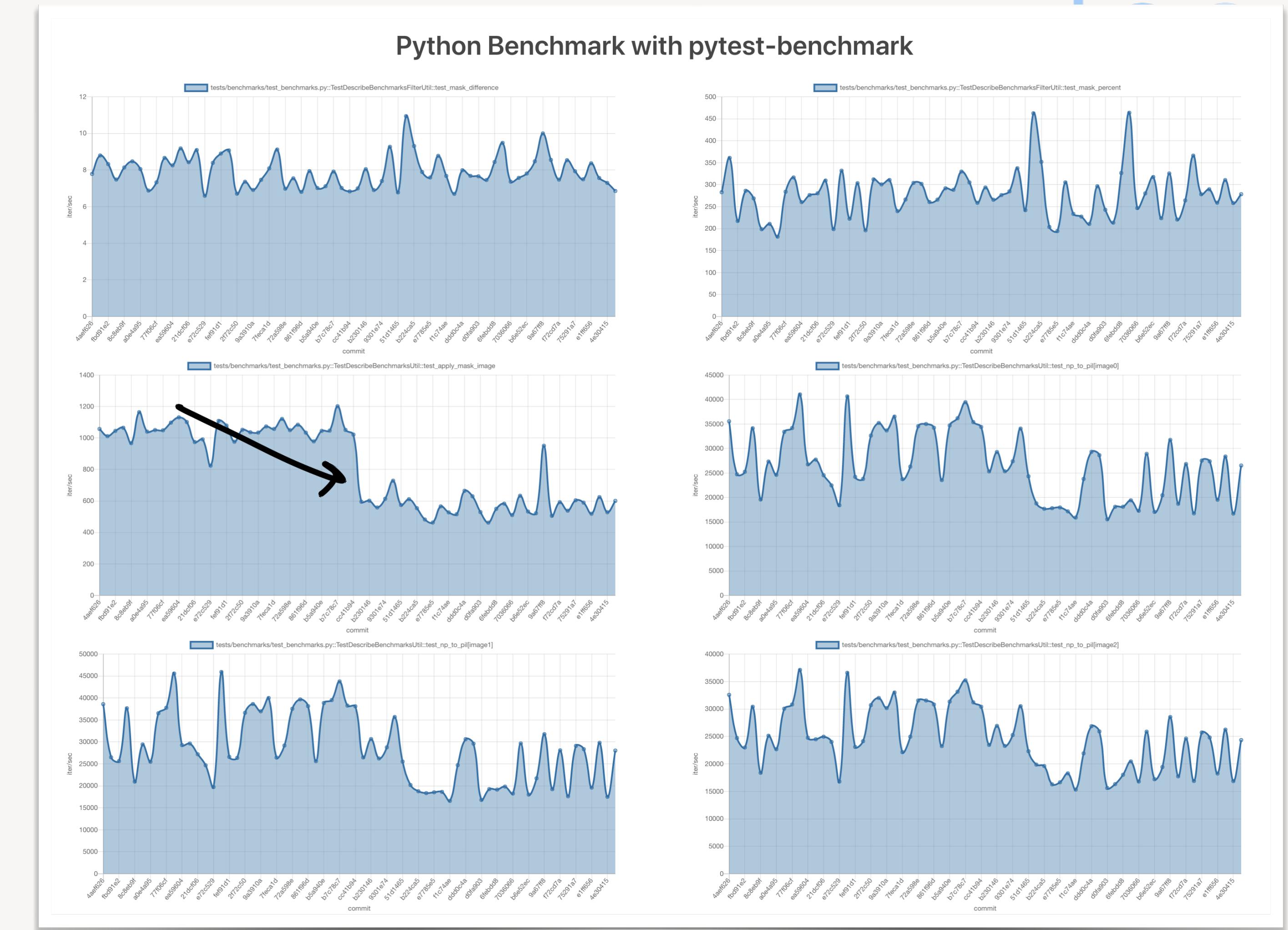
GitHub Actions  
for Continuous Integration

# benchmarks.yml

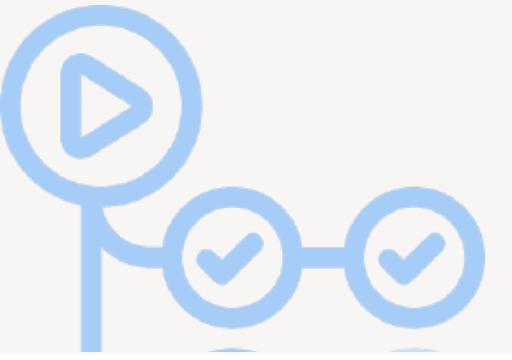
```

1 name: Python Benchmarks
2 on:
3   push:
4     branches:
5       - master
6 jobs:
7   benchmark:
8     runs-on: ubuntu-latest
9     if: ${{ github.repository }} == 'histolab/histolab'
10    steps:
11      - uses: actions/checkout@v2
12      - uses: actions/setup-python@v2
13      with:
14        python-version: '3.9'
15      - name: Run Benchmarks
16        run:
17          sudo apt install -y openslide-tools
18          pip install numpy pytest pytest-benchmark requests
19          pip install -e .
20          → pytest tests/benchmarks/test_benchmarks.py --benchmark-json output.json
21        # Download previous benchmark result from cache (if exists)
22        - name: Download previous benchmark data
23          uses: actions/cache@v1
24          with:
25            path: ./cache
26            key: ${{ runner.os }}-benchmark
27        - name: Store benchmark result
28          → uses: rhysd/github-action-benchmark@v1
29          with:
30            name: Python Benchmark with pytest-benchmark
31            tool: 'pytest'
32            output-file-path: output.json
33        # Personal access token to deploy GitHub Pages branch
34        github-token: ${{ secrets.PERSONAL_GITHUB_TOKEN }}
35        # Push and deploy GitHub pages branch automatically
36        auto-push: true
37        fail-on-alert: true
38        comment-on-alert: true
39        alert-comment-cc-users: '@ernestoarbitrio'
40        alert-threshold: '300%'

```



<https://histolab.github.io/histolab/dev/bench/>



# release.yml

```
1 name: Publish histolab to PyPI
2
3 on:
4   release:
5     types: [created]
6
7 jobs:
8   deploy:
9     runs-on: ubuntu-latest
10    steps:
11      - uses: actions/checkout@v2
12      - name: Set up Python
13        uses: actions/setup-python@v2
14        with:
15          python-version: 3.8
16      - name: Autobump version
17        run: |
18          # from refs/tags/v0.1.2 get 0.1.2
19          VERSION=$(echo $GITHUB_REF | sed 's#.*/v##')
20          PLACEHOLDER=$(grep '__version__ = ' histolab/__init__.py)
21          VERSION_FILE='histolab/__init__.py'
22          # ensure the placeholder is there. If grep doesn't find the placeholder
23          # it exits with exit code 1 and github actions aborts the build.
24          grep "$PLACEHOLDER" "$VERSION_FILE"
25          sed -i "s/$PLACEHOLDER/__version__ = \"${VERSION}\"/g" "$VERSION_FILE"
26        shell: bash
27      - name: Install dependencies
28        run: |
29          python -m pip install --upgrade pip
30          pip install setuptools wheel twine
31      - name: Build histolab
32        run: python setup.py sdist bdist_wheel
33      - name: Publish on test-PYPI
34        env:
35          TWINE_USERNAME: __token__
36          TWINE_PASSWORD: ${{ secrets.TEST_PYPI_APIKEY }}
37        run: twine upload --repository testpypi dist/*
38      - name: Publish on PYPI
39        env:
40          TWINE_USERNAME: __token__
41          TWINE_PASSWORD: ${{ secrets.PYPI_APIKEY }}
42        run: twine upload dist/*
```

histolab 0.3.0

pip install histolab

Released: Nov 6, 2021

Python library for Digital Pathology Image Processing

Navigation

Project description

Release history

Download files

Open Source

Project links

Homepage

Download

Statistics

GitHub statistics:

- Stars: 121
- Forks: 22
- Open issues/PRs: 24

View statistics for this project via [Libraries.io](#), or by using [our public dataset on Google BigQuery](#)

Test Status

CI passing codecov 100%

Code Quality

lgmt alerts 0 code quality: python A+ codemartr A code style black security bandit

Version Info

pypi v0.3.0 python 3.7 | 3.8 | 3.9 wheel yes

License

Apache-2.0

Documentation

docs passing

Compatibility Details

Operating System	Python version
Linux	3.7   3.8   3.9
MacOs	3.7   3.8   3.9

Publish on conda repositories CI enhancement

#348 opened 21 seconds ago by ernestoarbitrio

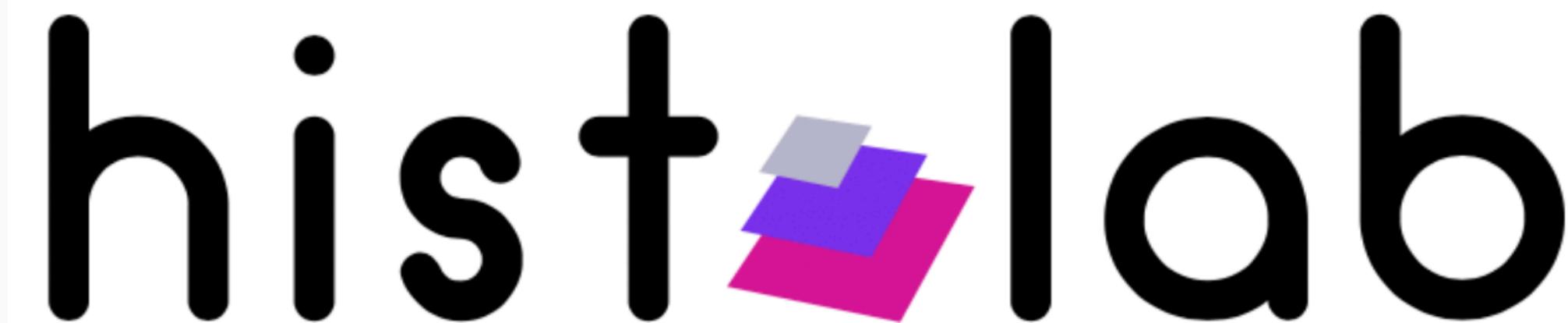
# DOCS

histolab.readthedocs.io



**SPHINX**  
PYTHON  
DOCUMENTATION  
GENERATOR

The screenshot shows the histolab documentation page. At the top, there's a header with the histolab logo and a search bar labeled "Search docs". Below the header is a sidebar with links to "Overview", "Installation", "Quick Start", "Define the tissue mask", "Contributing", and "Changelog". Under "API REFERENCE", there are links to "Slide", "Filters", "Masks", "Tile", "Tiler", "Scorer", "Data", and "Util". At the bottom of the sidebar is a call-to-action box with the text "Read the Docs for Business: Automated docs deployment of private & public repos. Get started today." and a "Read the Docs" button.



The aim of this project is to provide a tool for WSI processing in a reproducible environment to support clinical and scientific research. histolab is designed to handle WSIs, automatically detect the tissue, and retrieve informative tiles, and it can thus be integrated in a deep learning pipeline.

## Motivation

The histo-pathological analysis of tissue sections is the gold standard to assess the presence of many complex diseases, such as tumors, and understand their nature.

In daily practice, pathologists usually perform microscopy examination of tissue slides considering a limited number of regions and the clinical evaluation relies on several factors such as nuclei morphology, cell distribution, and color (staining): this process is time consuming, could lead to information loss, and suffers from inter-observer variability.

The advent of digital pathology is changing the way pathologists work and collaborate, and has opened the way to a new era in computational pathology. In particular, histopathology is expected to be at the center of the AI revolution in medicine [1], preision supported by the increasing success of deep learning applications to digital pathology.

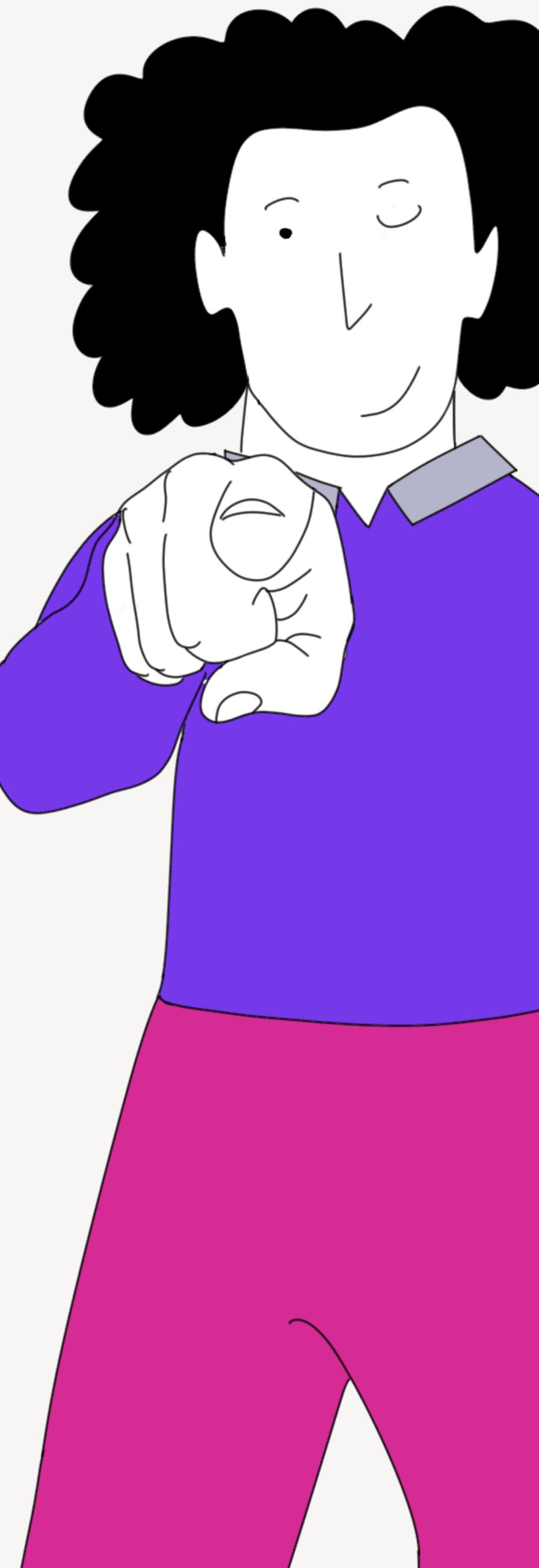
Whole Slide Images (WSIs), namely the translation of tissue slides from glass to digital format, are a great source of information from both a medical and a computational point of view. WSIs can be coloured with different staining techniques (e.g. H&E or IHC), and are usually very large in size (up to several GB per slide). Because of WSIs typical pyramidal structure, images can be retrieved at different magnification factors, providing a further layer of information beyond color.

However, processing WSIs is far from being trivial. First of all, WSIs can be stored in different

- Publish on conda repositories CI enhancement  
#348 opened 21 seconds ago by ernestoarbitrio
- Loading MRXS files have rendering problem. bug  
#341 opened 8 days ago by NeelKanwal
- wsi-artifacts job in CI taking too long enhancement help wanted  
#340 opened 8 days ago by alessiamarcolini
- Stain Normalization: Reinhard enhancement  
#322 opened on Sep 28 by alessiamarcolini ⇨ 0.5.0
- Expected bug - SlideSet not passing along args to Slide bug roadmap  
#307 opened on Aug 7 by kheffah ⇨ 0.4.0
- Better upsampling and downsampling modes best practice good first issue hacktoberfest  
#302 opened on Jul 25 by kheffah
- Extract the regions of interests from the WSI enhancement  
#298 opened on Jul 22 by suke18
- RandomTiler tiles have grid-like appearance in case of a very big image bug  
#295 opened on Jul 21 by alessiamarcolini
- Stain normalization: Macenko  
#288 opened on Jul 19 by alessiamarcolini ⇨ 0.5.0
- Add stain normalization filters enhancement  
#287 opened on Jul 19 by nicolebussola ⏺ 8 tasks ⇨ 0.5.0
- ScoreTiler heatmap scores enhancement  
#253 opened on Apr 18 by ernestoarbitrio
- TILAB Score dependencies enhancement  
#252 opened on Apr 18 by ernestoarbitrio
- Add non-overlapping option in Random Tiler enhancement  
#247 opened on Apr 12 by nicolebussola
- Support Python 3.8 under Windows enhancement idle  
#223 opened on Feb 8 by alessiamarcolini

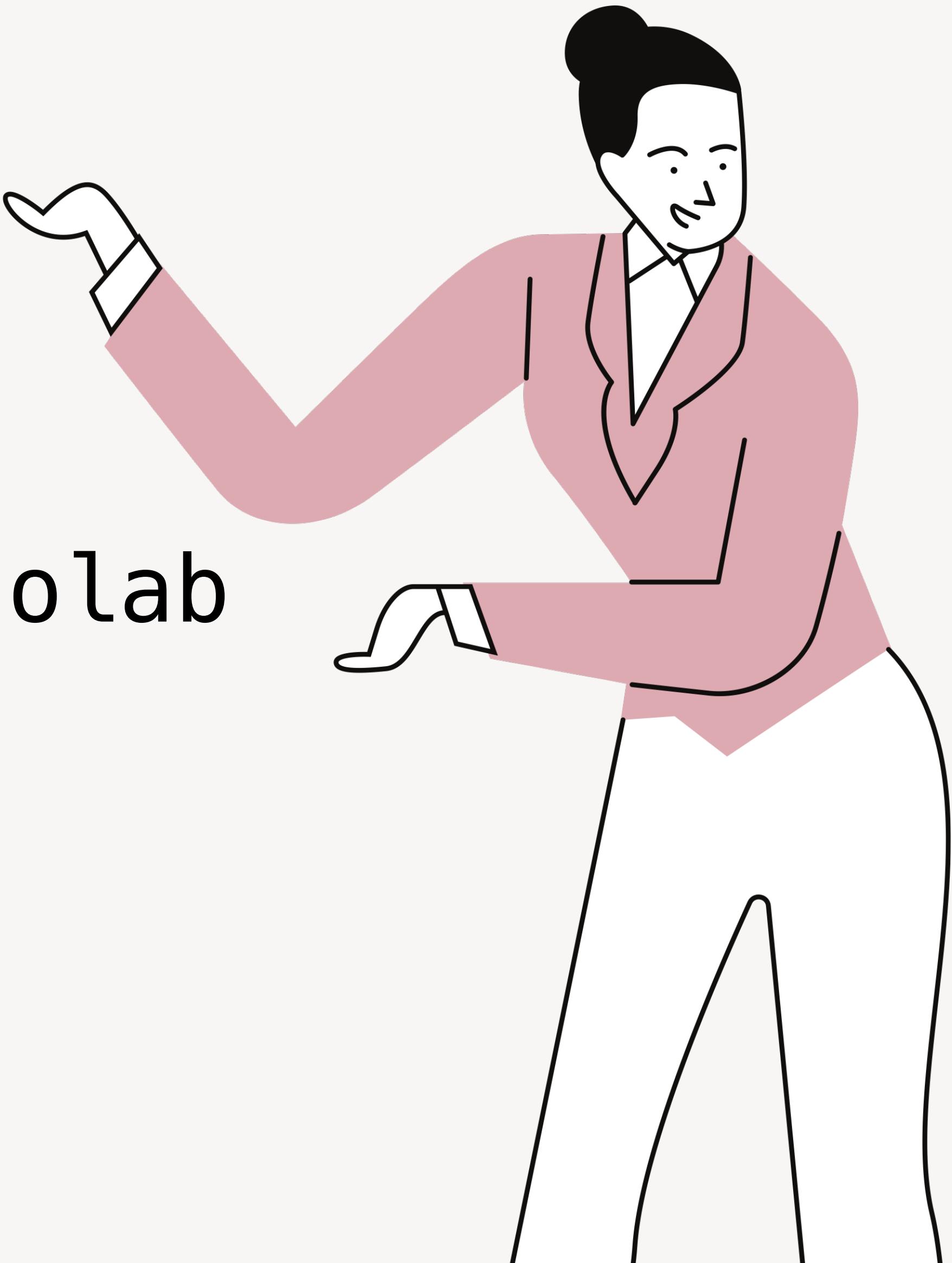
# Join hist<sup>2</sup>lab

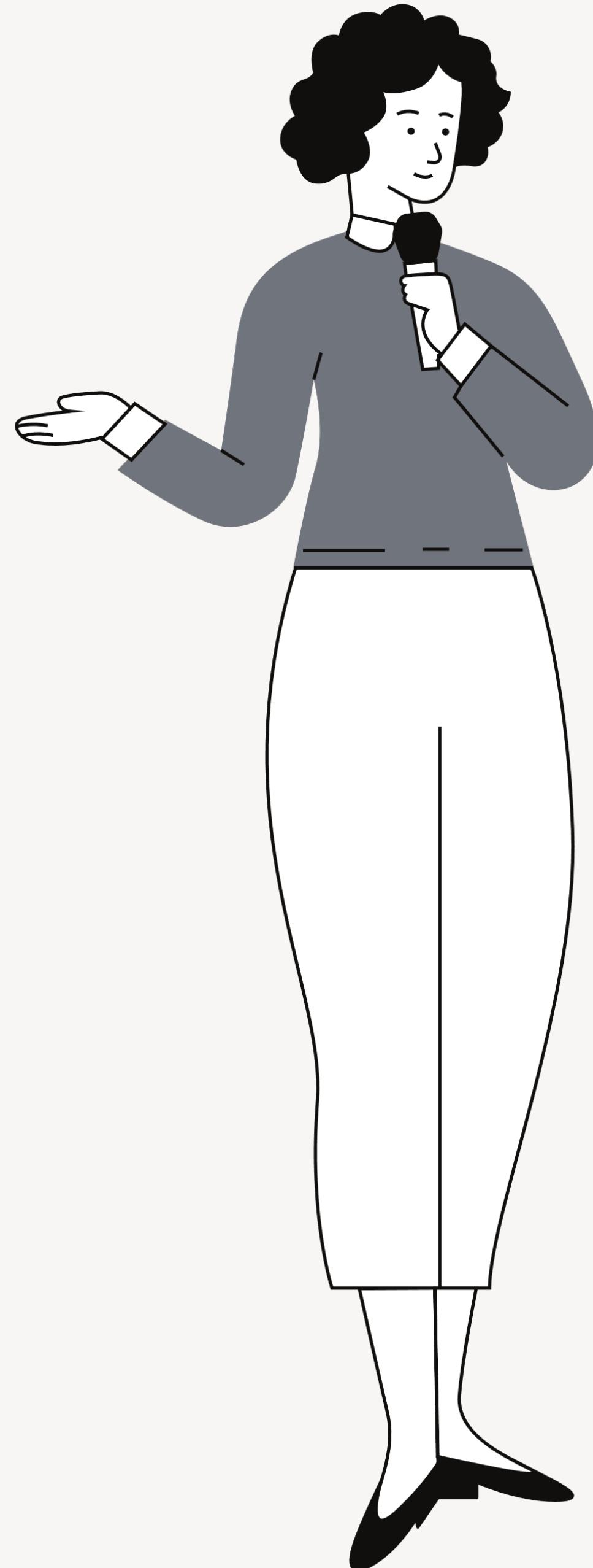
Give us a !



# Installation

```
$ pip install histolab
```





# Lessons learned and notes for future self

You will spend more time writing tests than code  
but at least you will sleep tight at night

100% coverage doesn't imply 0% bugs  
but stupid mistakes are easily caught

Code formatting and linting is nice  
but automatize it to focus on the important stuff

# Thank you

any question?

Alessia Marcolini

[alessia.marcolini@hk3lab.ai](mailto:alessia.marcolini@hk3lab.ai)

 @viperale

Ernesto Arbitrio

[ernesto.arbitrio@gmail.com](mailto:ernesto.arbitrio@gmail.com)

 @\_pamaron\_

Histolab is a joint work with



Nicole Bussola, PhD student  
@ FBK-MPBA / CIBIO