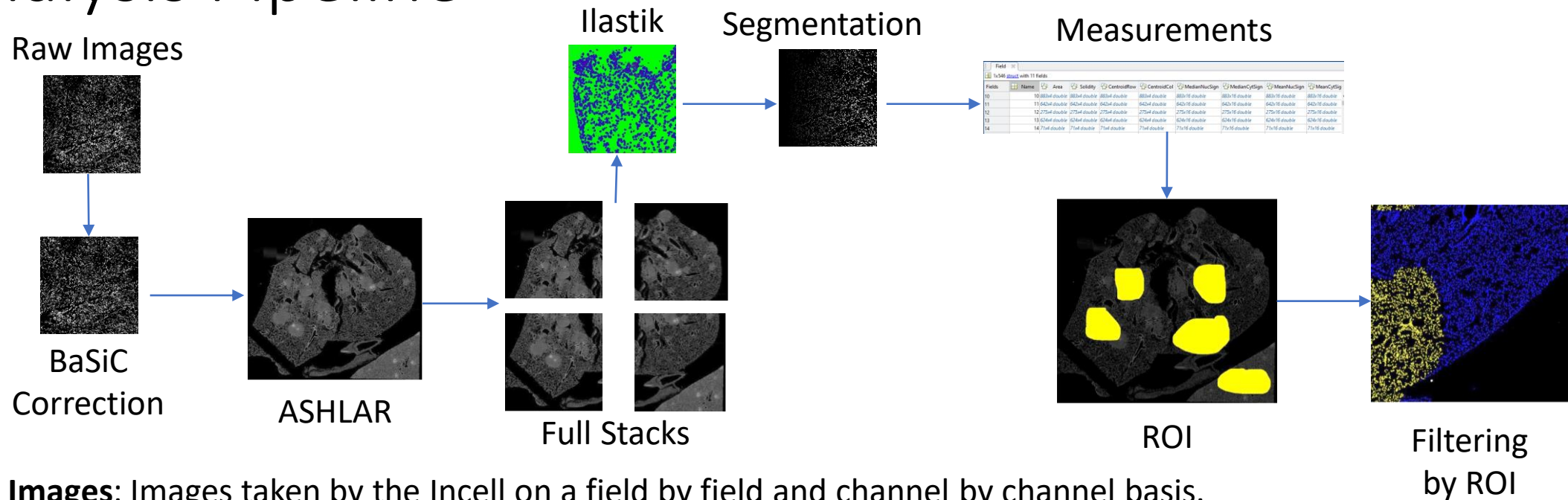


Whole Tissue Analysis Pipeline

03/04/19

Carmen Li

Analysis Pipeline



Raw Images: Images taken by the Incell on a field by field and channel by channel basis.

BaSiC Correction: Background and shading correction

ASHLAR: Stitches and registers each field into a whole image.

Full Stacks: Cuts the stitched and registered image into fields of a size specified by user and creates a stack of all cycle images. Cuts smaller images in each field for training Ilastik. Saves a small montage of DAPI first cycle for ROIs.

Ilastik: User trains Ilastik using a subset of the data. Ilastik produces segmentation probabilities.

Segmentation: Segments the nuclei based on the segmentation probabilities produced by Ilastik.

Measurements: Various measurements are taken for each antibody signal.

ROI: User draws ROI on montage in ImageJ

Filtering by ROI: Gets indexes of cells within the ROI. Measurement data can be filtered based on ROI.

Github repository: <https://github.com/giorgiogaglia/CyclF-Pipelines/tree/master/WholeTissueAshlaredPipeline>

Codes:

1. **Basic Correction and ASHLAR** ([run_ashlar_batch_csv_incell.py](#); Command Line): Background and shading correction and stitching and registering each field into a whole image
2. **Script to run Step 3, 5 and 6** ([RUN_All.m](#); **MATLAB**): Script to run all the MATLAB scripts
3. **Preprocessing before Ilastik:**
 - a) **Full Stacks and Cropped Stacks** ([RUN_Step1_new_fields_preilastik_crops.m](#); **MATLAB**): Cuts fields to a size specified by the user and cuts out smaller cropped images in each field for training in Ilastik. Saves a small montage of the DAPI of the first cycle for ROIs.
 - b) **Processing stacks for Ilastik** ([changetifffromZtoXchannel_nocrops.ijm](#); **ImageJ**): Changes properties of the stack from Z to channels. Required for Ilastik.
4. **Ilastik** ([GUI or headless on O2](#)): User trains Ilastik using cropped stacks. Ilastik produces segmentation probabilities.
5. **Segmentation** ([RUN_Step3_segmentfromilastik.m](#); **MATLAB**): Segments the nuclei based on segmentation probabilities produced by Ilastik
6. **Measurements** ([RUN_Step4_CyclF_measurements_ilastik.m](#); **MATLAB**): Takes various measurements for each antibody signal.
7. **ROI** ([RUN_Step5_ROI.m](#); **MATLAB**): User draws ROIs on ImageJ. Indexes of each pixel in the ROI are found. Indexes of centroids in the ROI are also found.

How to run ASHLAR

- Requirements:
 - “run_ashlar_batch_csv_incell.py” file created by YuAn
 - <https://gist.github.com/Yu-AnChen/f9030c9d6425cb293846a729d1b224ed>
 - CSV file:
 - First column: Directory (location of folder that holds all the raw data for the slide)
 - Second column: Correction (y/n; Basic Correction)
 - Third column: Pyramid (y/n)
- Open command prompt:
 - D: (directory that the script is in)
 - cd D:\path\ (folder that the script is in)
- Command line: `python run_ashlar_batch_csv_incell.py Nameofcsv.csv`

RUN_ALL.m

- What it does:
 - Runs step 1,3 and 4 after ASHLAR: Fullstacks, Segmentation and measuring
- User input:
 - Location of ASHLARed image
 - Name of ome.tif w/o “ome.tif” ending
 - Size of field desired (field will be a square)
 - Number of crops desired
 - Number of cycles
 - Location of folder that contains Ilastik probabilities and Full stacks
 - Channels corresponding to Ilastik probabilities
 - Foci segmentation parameters

RUN_Step1_new_fields_preilastik_crops.m

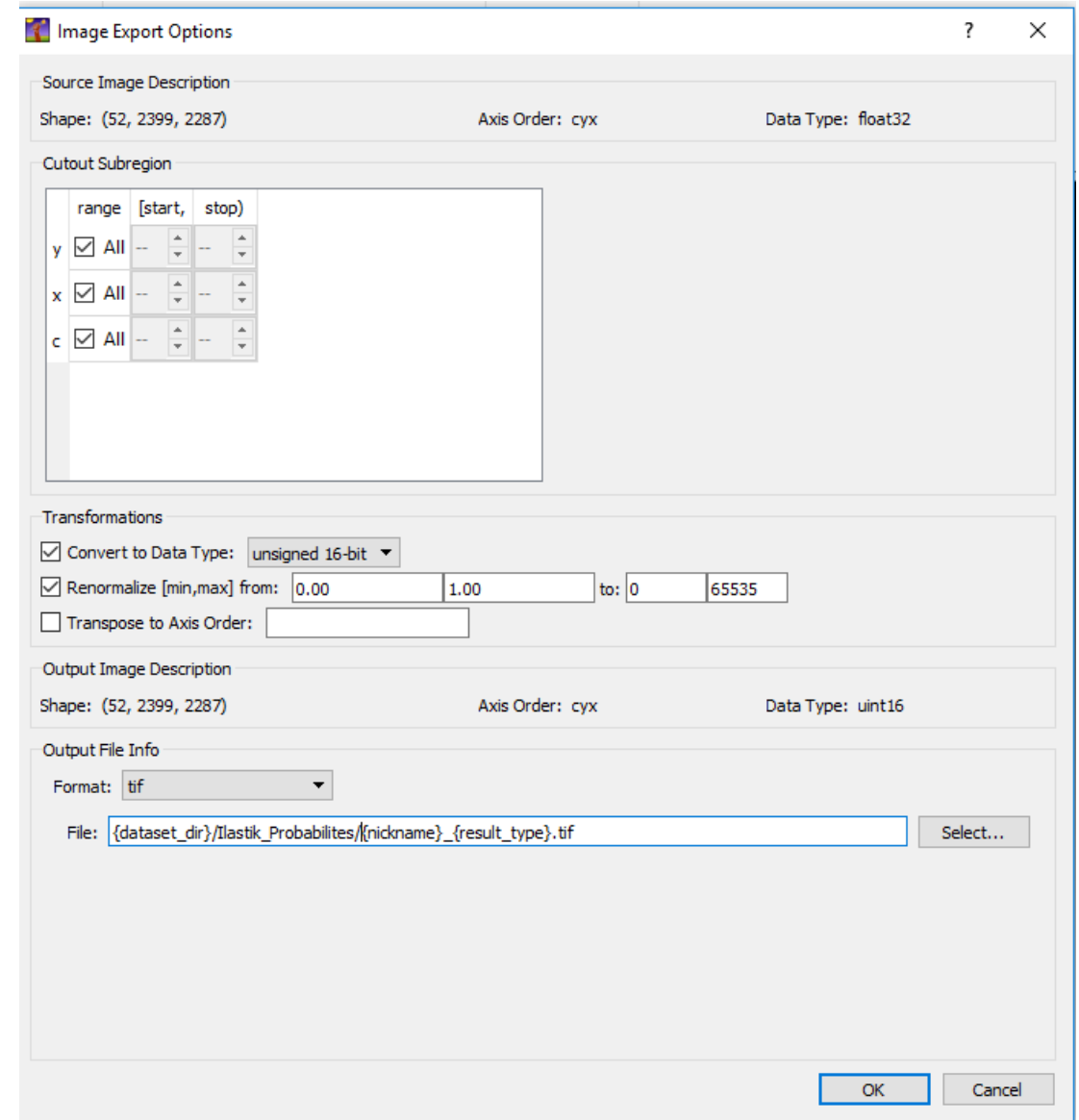
- What it does:
 - Cuts ASHLARed image into fields of the size specified by user
 - Creates stack of all channels for fields that contain cells
 - Cuts cropped images of 250X250 pixels from the field, for Ilastik training
 - Creates stack of all channels for the cropped image
- Inputs:
 - Location of ASHLARed image
 - Name of ome.tif w/o “ome.tif” ending
 - Size of field desired (field will be a square)
 - Number of crops desired
 - Number of cycles
- Outputs:
 - NameofOmetif_Field_row_column.tif: stack of all channels
 - Coordinates_Fullstacks\NameofOmetif.mat:
 - Matrix of [if field is kept(1/0), x0,x1,y0,y1]
 - NameofOmetif_Field_row_column_crop#.tif: stack of all channels for cropped image
 - NameofOmetif_montage.tif

changetifffromZtoXchannel_nocrops.ijm

- What it does:
 - Changes the slices from Z to X for Ilastik processing
- User Input:
 - Number of channels
 - Location of full stacks and cropped images
- Output:
 - Full stack and cropped image tiff with Z changed to X

Ilastik

- Create project
- Input cropped images for training
- Train using three qualifications: nucleus, cytoplasm, background
- Input fields
- IMPORTANT: Export settings should be:
 - Export Source: Probabilities
 - Cover to Data Type: unsigned 16-bit
 - Renormalize [min,max] from: 0.0-1 .0 to 0:65535
 - Filename_format:
**{dataset_dir}/Ilastik_Probabilities/{nickname}_P
robabilities.tif**
 - File_format: tiff
- Use batch processing to create segmentation probabilities



RUN_Step3_segmentfromilastik.m

- What it does:
 - Takes in Ilastik probabilities and produces segmentation
- User input:
 - Ilastik probabilities
 - FullStacks
- Output:
 - Segmented image
 - Check segmented image

RUN_Step4_CyclF_measurements_ilastik.m

- What it does:
 - Takes measurements of the signal of the antibody
- User Input:
 - Segmented Image
 - FullStacks
- Output:
 - NameofOmeTif_Results_date.mat: matrix of measurements
 - Nucleus and cytoplasm segmented image
 - Foci segmented image

RUN_Step5_ROI.m

- What it does:
 - Finds the pixels in the ROI area
- User Input:
 - ROI drawn on ImageJ
 - Montage that ROI was drawn on (Created in RUN_Step1...)
 - Coordinates of each field
- Output:
 - ROI_pixels.mat: matrix of pixel locations (x,y) of ROIs, vector of **row** indexes of Centroids that are within the ROI

Other Scripts

Might be useful, might not be useful. All on Github

RUN_Step6_PlottingResults.m

- Example of how you would use indexes to filter measurements in mat file created in RUN_Step4

renamefiles_v2.m

- Renames multiple files
- In Other Scripts folder of Github

Ilastik on O2

Need to Download and Install

- Filezilla: to upload to O2
- <https://filezilla-project.org/download.php>
- MobaXterm: to access O2 on Windows
- <https://mobaxterm.mobatek.net/download.html>

Installing Linux version of O2

mfk8@login02:~\$ srun --pty -p interactive -t 0-12:00 /bin/bash (Interactive node)

mfk8@compute-a-01-01:~\$ wget <http://files.ilastik.org/ilastik-1.3.2rc2-Linux.tar.bz2>

mfk8@compute-a-01-01:~\$ tar -vxjf ilastik-1.3.2rc2-Linux.tar.bz2

IMPORTANT: Make sure you are downloading the version you need

<http://files.ilastik.org/>

Example script to run 1 Field

IMPORTANT: Must have training set of data in same folder format for Ilastik to work

SH Script:

```
#!/bin/bash
#SBATCH -p short (Type of Partition)
#SBATCH -t 0-03:00 (Time requested)
#SBATCH --mem=100G (Memory Requested)
#SBATCH -o %j.out (Out file naming: JobID..out)
./ilastik-1.3.2rc2-Linux/run_ilastik.sh \ (Calling Ilastik)
--headless \ (Non GUI mode)
--project=/n/scratch2/cl401/MouseLung/2019-01-03_MouseLungTumors_training_headless.ilp \ (Project)
--export_source="Probabilities" \ (Export type)
--export_dtype=uint16 \ (Data type)
--pipeline_result_drange="(0.0,1.0)" \ (Import data range)
--export_drange="(0,65535)" \ (Export Data range)
--output_format=tif \ (Output file type)
--output_filename_format={dataset_dir}/Ilastik_Probabilities/{nickname}_Probabilities.tif \ (output filename format: location)
/n/scratch2/cl401/MouseLung/AJ0176_P2_Field_02_01.tif (File to be run through ilastik)
```

How to create a script

- vim scriptname.sh
- Type **i** for insert mode
- Start typing
- When finished, press **ESC** key
- To save, type **:wq**, then press **Enter** key
- To run the script: sbatch Scriptname.sh

How to check on the job you are running:

`queue -u UserName` (Shows Job id and information)

`watch queue -u UserName` (updates the queue every 2 seconds)

`sacct` (Shows list of jobs you are running/ have run for the day)

`tail -f JobID.out` (Shows last ten lines of “out” file)

`less JobID.out` (shows entire “out” file)

`O2sacct JobID` (returns information about the job after completion)

Example script to run Ilastik on multiple files

To create script: vim NameofScript.sh

To save and exit out of writing script: Shift + Z + Z

for f in {Folder w/ all files}/*.tif; do

cmd="./ilastik-1.3.2-Linux/run_ilastik.sh \

--headless \

--project=./ilastik-1.3.2-Linux/2019-01-03_MouseLungTumors_training_headless.ilp \ (Project)

--export_source=Probabilities \ (Export type)

--export_dtype=uint16 \ (Data type)

--pipeline_result_drange='(0.0,1.0)'

--export_drange='(0, 65535)' \ (Data range)

--output_format=tif \ (Output file type)

--output_filename_format={dataset_dir}/Ilastik_Probabilities/{nickname}_Probabilities.tif \ (Output file format:
location)

\$f" (File that is run in Ilastik)

sbatch -p short -t 0-01:00 --mem=100G -o %j.out --wrap="\$cmd"

done

Completed ilastik 1 field (6000X6000 pixels)

- Time = 01:3:03
- Memory used = 72.01G

35756800.ba+	COMPLETED	compute-e-16-245	2019-03-
05T15:37:10	01:03:30	01:03:30	01:02:46
cpu=1,mem=100G,node=1			72.01G

O2 Tips and tricks

CTRL + L (clear screen)

CTRL + A (jumps to beginning of line)

rsync -a (command to upload data; -a: uploads metadata)

Slurm= 10TB size

Home= 100GB size

Helpful Links

<https://wiki.rc.hms.harvard.edu/display/O2/Local+Software+Installs>

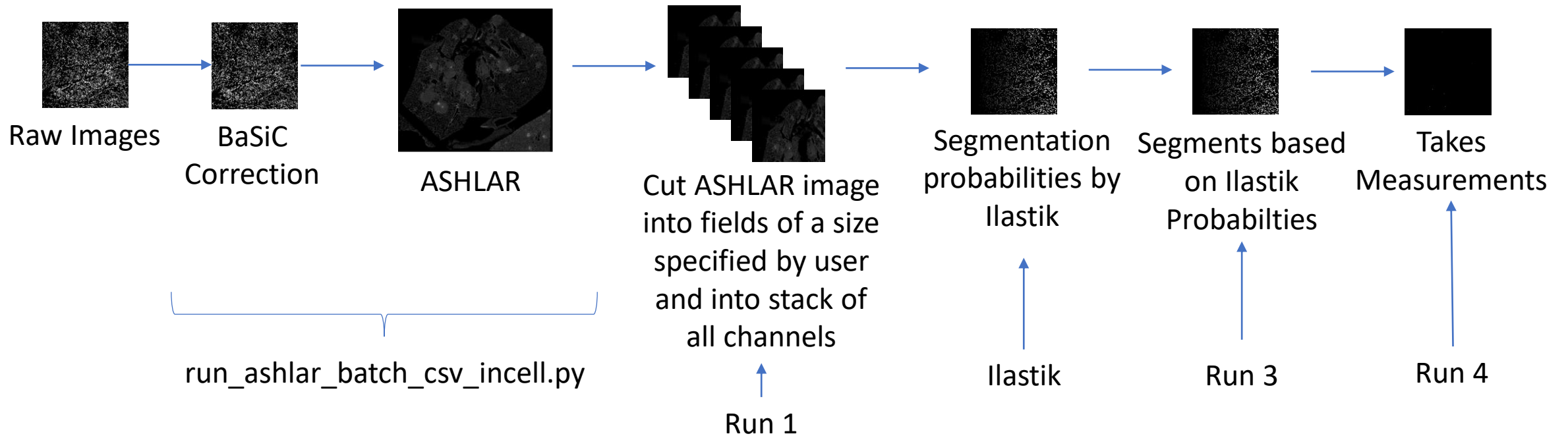
<https://www.ilastik.org/documentation/basics/headless>

Old slides + extra stuff

Headless on windows

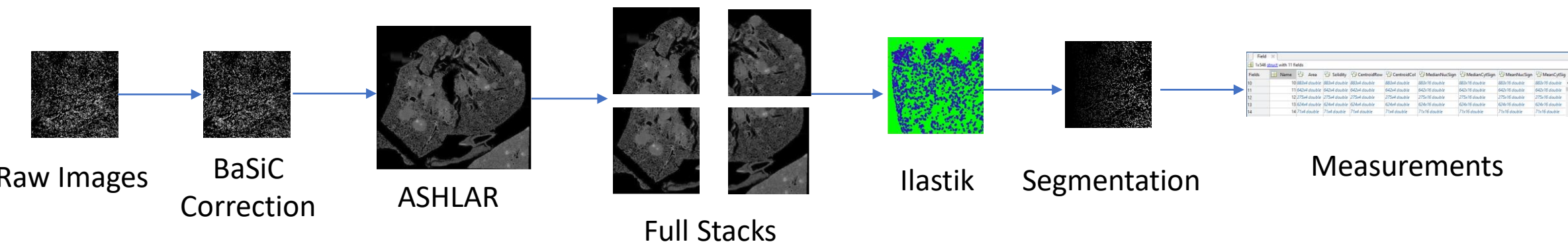
```
.\run-ilastik.bat --headless --  
project=Z:\sorger\data\IN_Cell_Analyzer_6000\Giorgio\2019-01-  
03_Mouse_Lung_Tumors_Round2\Analysis\2019-01-  
03_MouseLungTumors_training.ilp --export_source="Probabilities" --  
export_dtype=uint16 --pipeline_result_drange="(0.0,1.0)" --  
export_drange="(0,65535)" --output_format=tif --  
output_filename_format={dataset_dir}/Ilastik_Probabilities/{nickname  
}_Probabilities.tif  
Z:\sorger\data\IN_Cell_Analyzer_6000\Carmen\O2test\Field_01_03_1  
7.tif
```

Analysis Pipeline

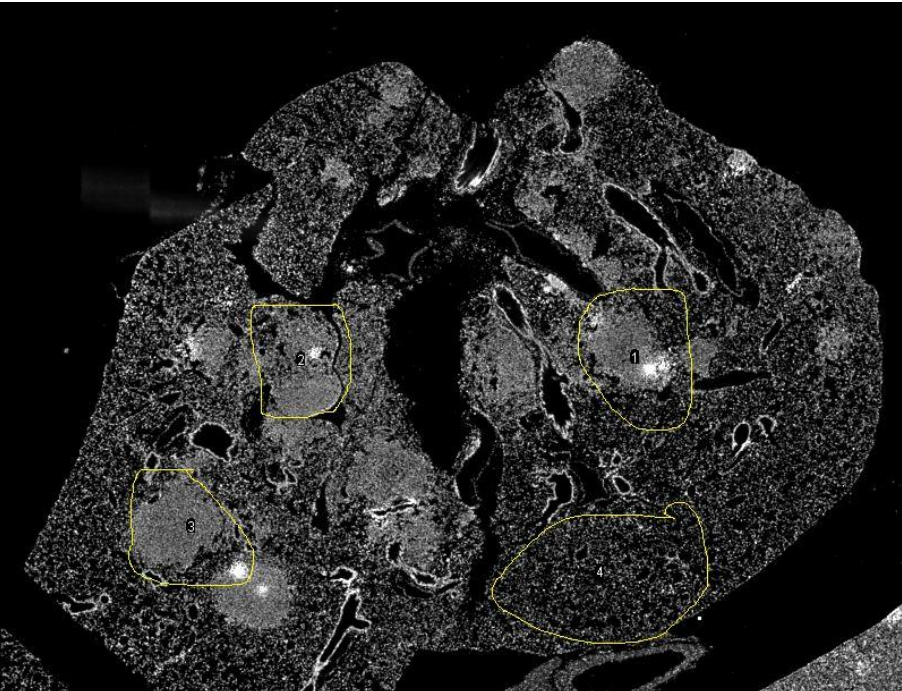


Github repository: <https://github.com/giorgiogaglia/CyclF-Pipelines/tree/master/WholeTissueAshlaredPipeline>

Analysis Pipeline



Raw Images
BaSiC Correction
ASHLAR: Still
Full Stacks:
images. Cut
Ilastik: User
Segmentation
Measurements
Github repo



and channel by channel basis.

age.

ds of a size specified by user and create
k. Saves a small montage of DAPI first cy
produces segmentation probabilities.
ation probabilities produced by Ilastik.
antibody signal.

[Pipelines/tree/master/WholeTissueAsh](#)

