



Batch PIV in Chile

Transferring terabytes of data turns out to be very difficult. In order to ensure unified PIV treatment on all the images, I will write an easy to use script and let Cristian run on his local data. The job should be a faithful replicate of what I have done in Paris. Below is a step-by-step guide.

0. Install Miniconda

- [Miniconda download page](#)
- open windows command prompt (cmd)
- create a virtual environment:

```
conda create -n piv python=3.9
```

When asked `Proceed ([y]/n)?`, say `y`.

- activate the environment:

```
conda activate piv
```

1. Install libraries and scripts

First, install `myimagelib` package, which contains many utility functions I've written over the years:

```
pip install myimagelib --upgrade
```

Then, unzip all the scripts to a local folder. Here's a checklist of the scripts:

- ☐ `gen_preview.py`
- ☐ [PIV.py](#)
- ☐ `apply_mask.py`
- ☐ `wrap_piv.py`
- ☐ `batch_gen_preview.py`
- ☐ [run.py](#)

2. Prepare data

All the nd2 videos should be put in the same folder, say `nd2`. The expected folder structure is

```

|-- nd2
    |-- 000.nd2
    |-- 001.nd2
    |-- ...
    |-- log.csv
    --- (The following directories will be generated by the scripts) ---
    |-- preview
        |-- 000.tif
        |-- 001.tif
    |-- mask
        |-- 000.tif
        |-- 001.tif
    |-- PIV
        |-- 000.mat
        |-- 001.mat

```

Initially, we have only nd2 files in the folder nd2. Note that we rename all the nd2 files as {DE#}.nd2 , where DE# is a 3-digit integer, with leading space filled with 0's.

batch_gen_preview.py will produce folder preview . Then duplicate the preview folder and rename as mask . Then use ImageJ to draw binary masks based on the preview images.

Then run.py will perform PIV analysis on all the nd2 videos, and create PIV folder.

The log.csv file is sent with the scripts, please put it also in the nd2 folder, so that the script can locate it correctly.

3. Generate preview

In your commandline tool (cmd), cd to the directory where the scripts are saved. For example, on my computer it is C:\Users\liuzy\Documents\Github\script .

```

(testpip) C:\Users\liuzy\Documents\Github\script>dir
Volume in drive C is OS
Volume Serial Number is FE22-B21F

Directory of C:\Users\liuzy\Documents\Github\script

01/05/2023  11:32 AM    <DIR>          .
01/05/2023  11:32 AM    <DIR>          ..
12/08/2021  11:21 AM                66 .gitattributes
11/16/2022  10:00 AM    <DIR>          .github
01/03/2023  01:49 PM                1,561 .gitignore
11/03/2022  03:42 PM    <DIR>          .ipynb_checkpoints
12/06/2022  02:09 PM    <DIR>          .vscode
01/05/2023  02:34 PM                1,521 apply_mask.py
03/16/2022  01:12 PM                6,839 Archive.zip
01/03/2023  02:27 PM                1,916 batch_gen_preview.py
11/15/2022  04:33 PM                1,617 batch_mean_velocity.py
11/15/2022  04:34 PM                1,599 batch_spatial_correlation.py

```

Then run the following command:

```
python batch_gen_preview.py /your/folder/nd2
```

Here you need to specify the directory of the nd2 folder. In case you have white spaces in the directory, use full quotation marks (`" "`) to enclose the directory.

If the script executes correctly, you should see some info strings like:

```
(testpip) C:\Users\liuzy\Documents\Github\script>python batch_gen_preview.py D:\nd2
Thu Jan  5 15:58:01 2023
-----
Run batch_gen_preview on D:\nd2
Results will be saved in D:\nd2\preview
The following files will be processed:
  D:\nd2\023.nd2
  D:\nd2\024.nd2
  D:\nd2\025.nd2
  D:\nd2\030.nd2
  D:\nd2\035.nd2
  D:\nd2\037.nd2
```

4. Make masks

1. Duplicate the preview folder

preview	1/5/2023 3:58 PM	File folder	
preview - Copy	1/5/2023 4:01 PM	File folder	
023.nd2	1/3/2023 3:17 PM	ND2 File	7,264,040 KB
024.nd2	1/3/2023 3:41 PM	ND2 File	38,065,740 ...
025.nd2	1/4/2023 10:21 AM	ND2 File	38,065,740 ...
030.nd2	1/4/2023 10:42 AM	ND2 File	7,688,936 KB
035.nd2	1/3/2023 5:26 PM	ND2 File	21.839.740 ...

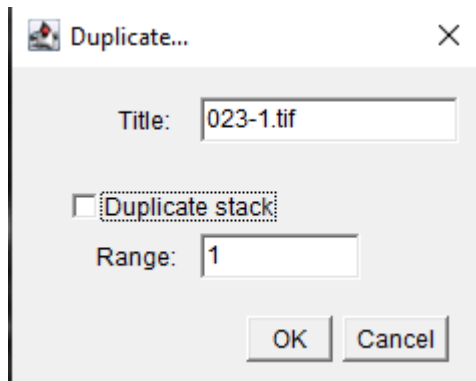
2. Rename it as `mask`

preview	1/5/2023 3:58 PM	File folder	
mask	1/5/2023 4:01 PM	File folder	
023.nd2	1/3/2023 3:17 PM	ND2 File	7,264,040 KB
024.nd2	1/3/2023 3:41 PM	ND2 File	38,065,740 ...
025.nd2	1/4/2023 10:21 AM	ND2 File	38,065,740 ...
030.nd2	1/4/2023 10:42 AM	ND2 File	7,688,936 KB

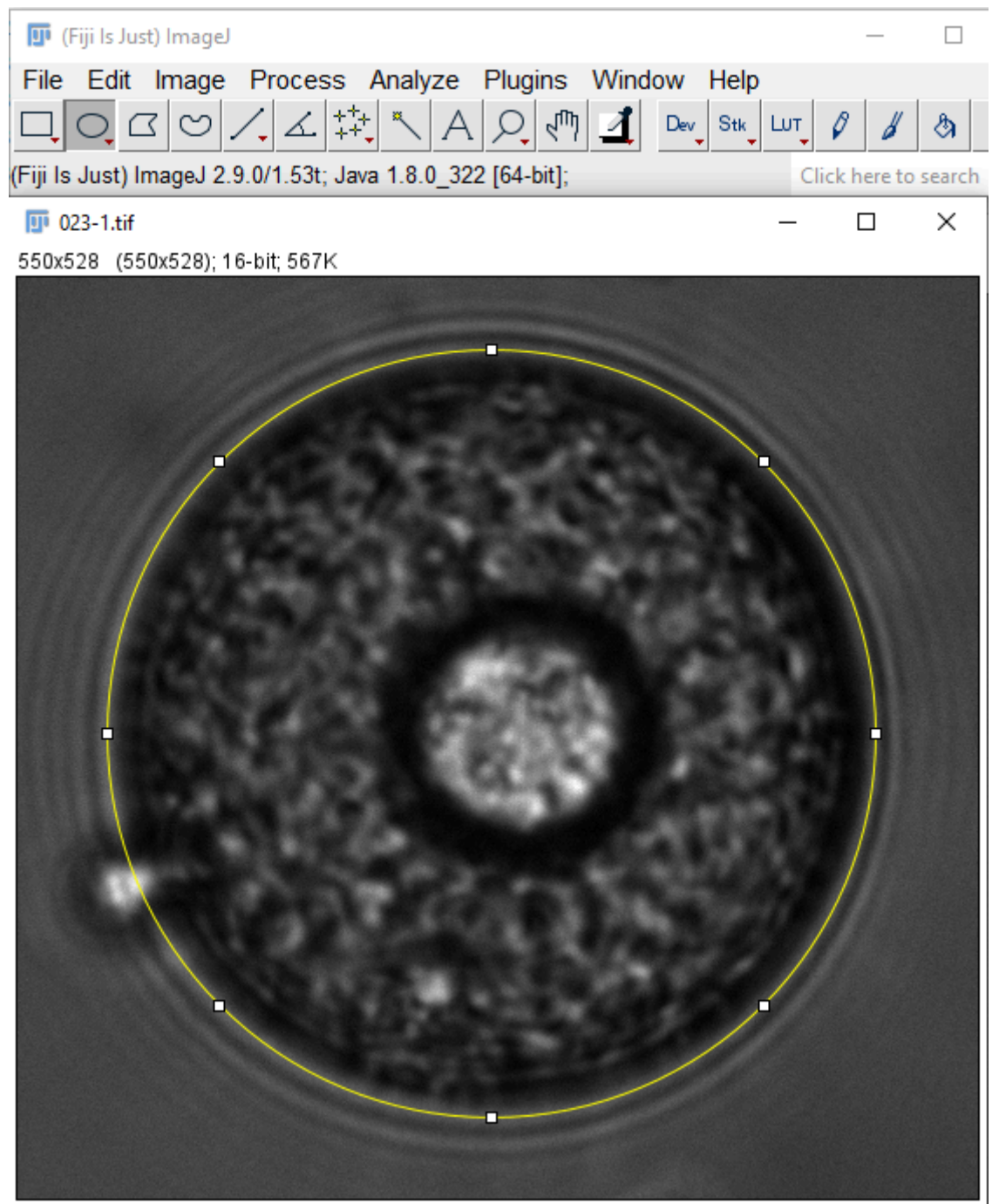
3. Enter `mask` folder, open the first tif with ImageJ

Name	Date modified	Type	Size
023.tif	1/5/2023 3:58 PM	TIF File	1,135 KB
024.tif	1/5/2023 3:58 PM	TIF File	2,482 KB
025.tif	1/5/2023 3:58 PM	TIF File	2,482 KB
030.tif	1/5/2023 3:58 PM	TIF File	2,482 KB
035.tif	1/5/2023 3:58 PM	TIF File	2,482 KB
039.tif	1/5/2023 3:58 PM	TIF File	5,068 KB
042.tif	1/5/2023 3:58 PM	TIF File	16,353 KB
044.tif	1/5/2023 3:58 PM	TIF File	1,782 KB

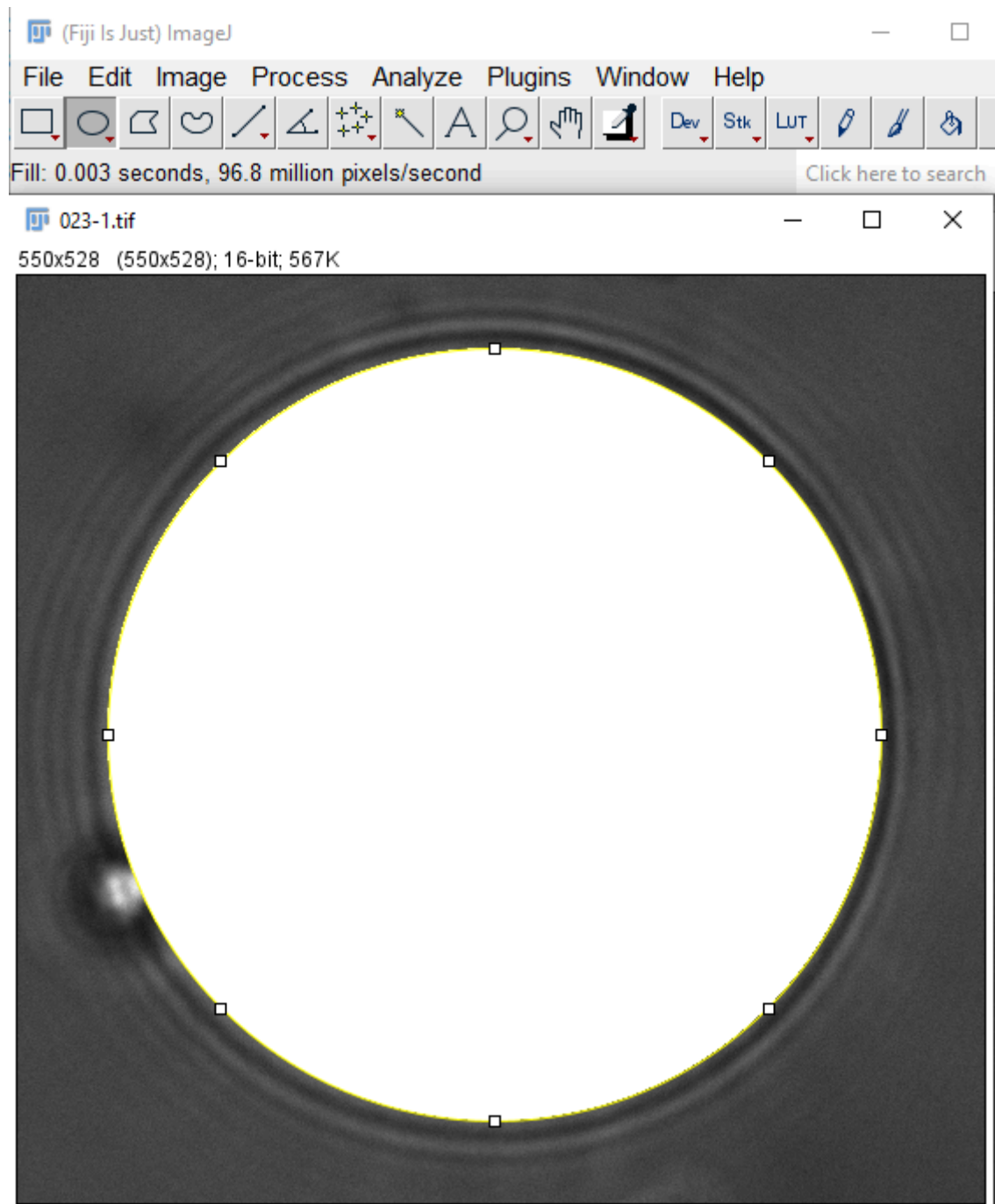
4. Check if the positions of outer droplet are consistent in the two frames (they are the first and the last frames of the original video).
5. If OK, duplicate the first frame with `Ctrl+Shift+D` , uncheck "Duplicate stack" and set range to 1.



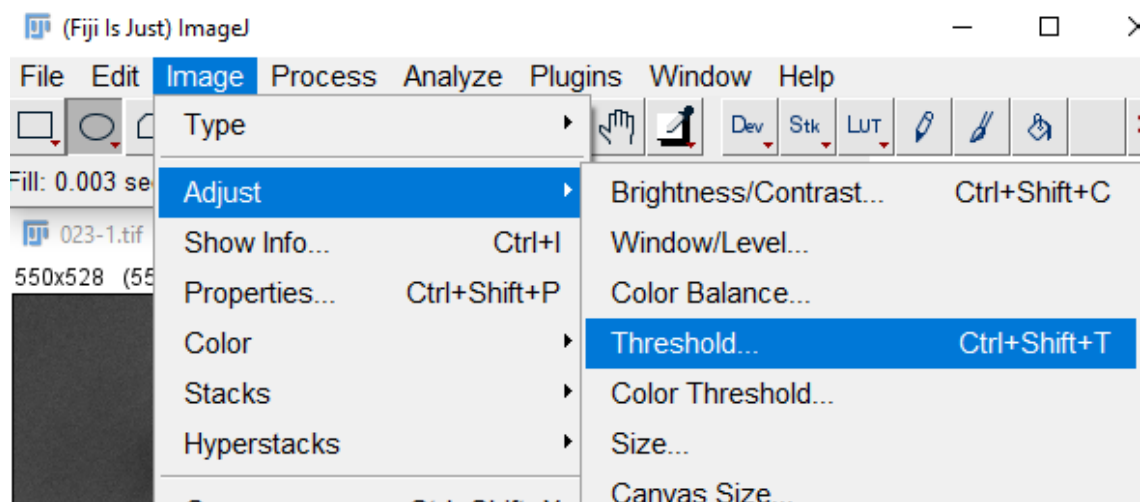
6. In the duplicated frame, use circle tool to select a circle that overlaps with the outer droplet edge.



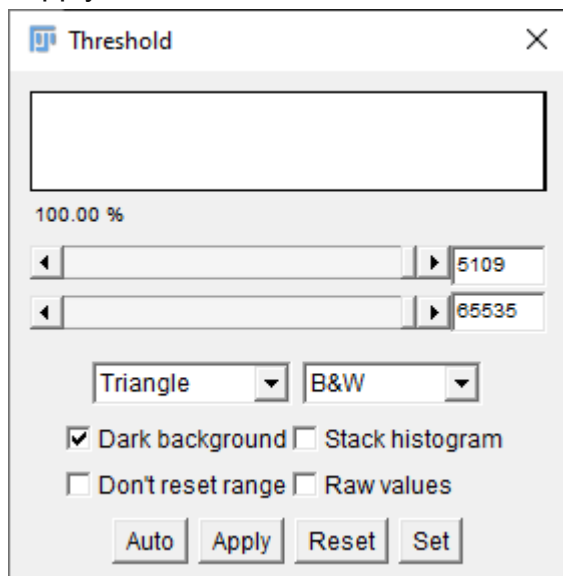
7. Fill the circle with white color (Ctrl+F)



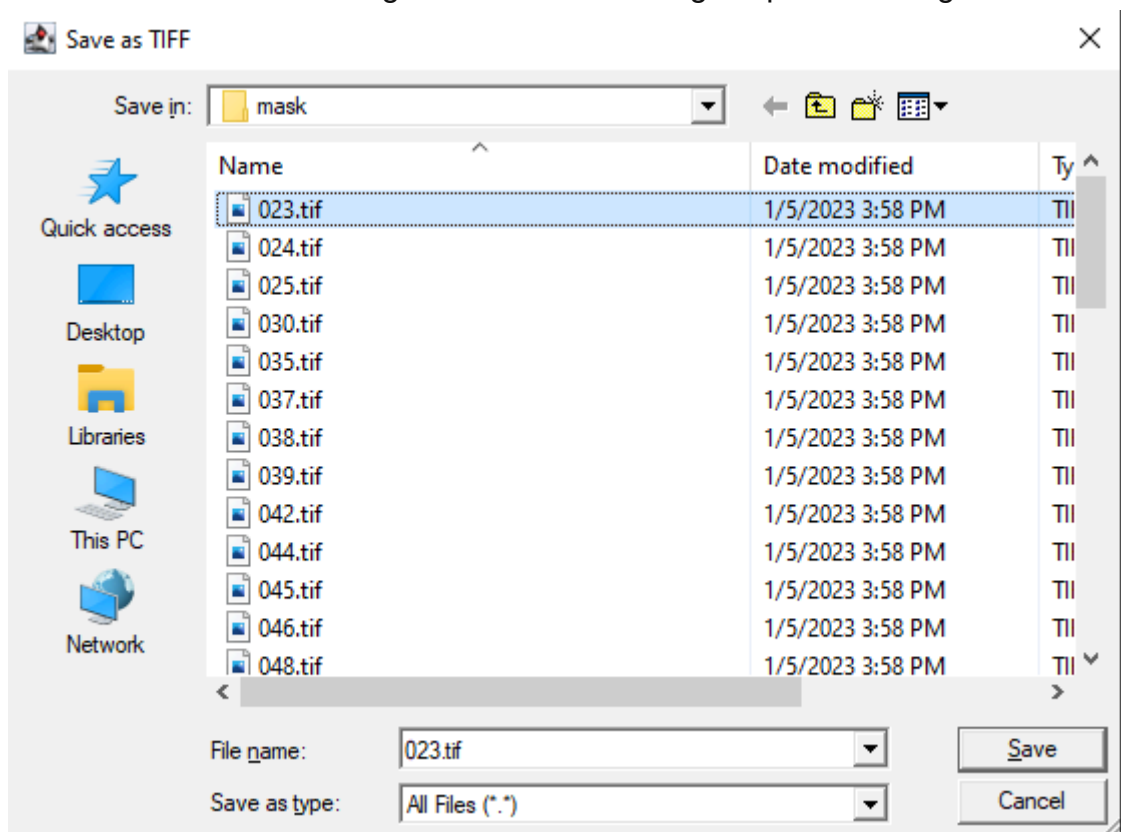
8. Open Image->Adjust->Threshold menu.



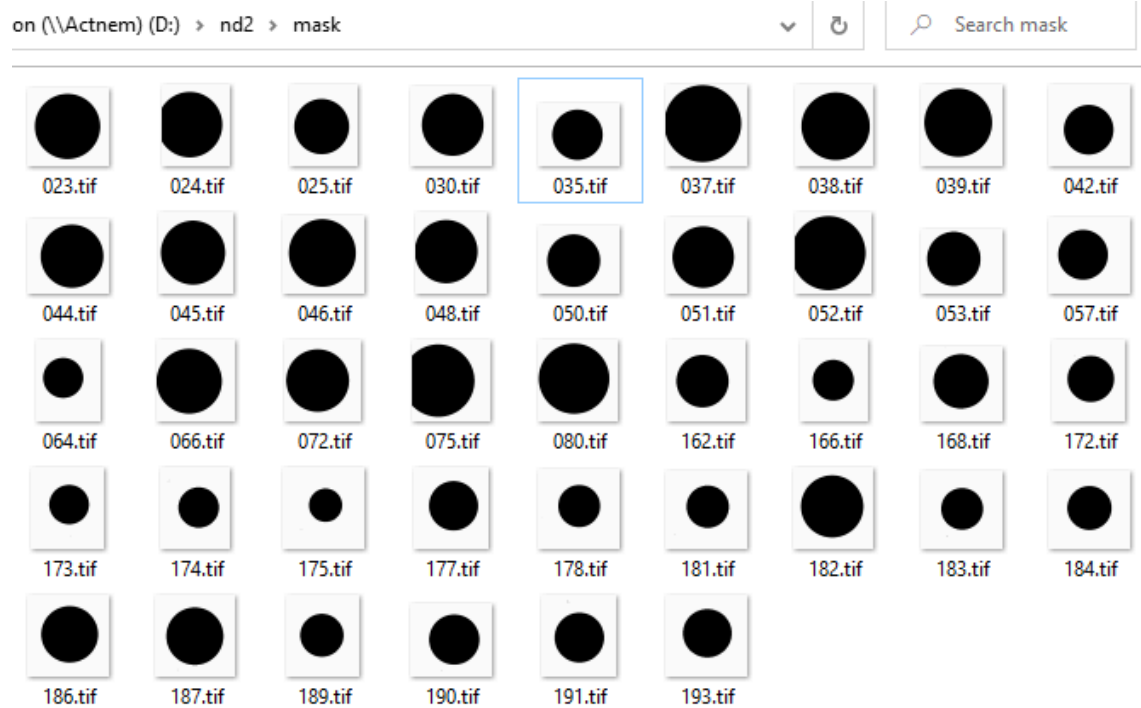
9. Set algorithm to "Triangle", check "Dark background", then click "Auto" and "Apply".



10. Double check if the droplet part has large values and the outside part has small values. Then save the image to overwrite the original preview image.



11. Repeat this procedure to all the previews. Eventually the mask folder will look like this:



5. Execute `run.py`

`run.py` consists of PIV, applying mask and wrapping PIV into compact form. It should be executed only after mask preparation (step 4), since masks are need for this script.

```
python run.py /your/folder/nd2
```

Once finished, you will have a `PIV` folder under your `nd2` folder, filled with `xxx.mat` files. Send them to me and I will do the downstream analysis. Also you are welcome to play around with the data!