Bring up report for Autocyplex.

# Orientation/System Overview

# Basic hardware Testing

## Illumination Profile

## 2.2 Resolution

## 2.3 Reflection

## 2.4 Invisislip vs water imaging comparison

## 2.5 Planar focal plane flatness

## 2.6 chromatic aberration

# Software

## Acquisition Engine

### V1.0

Capabilities entail:

* Hard drive saved focus map object
* Optional auto focus and auto exposure activation
* Acquisition order of ZC and XY in snake from upper left corner. Saves tiles every z stack
* ‘Wakes’ up light engine
* Dictate X dimensional frame size and adjusts focus map to compensate
* Hard drive storage of single cycle of exposure times

### V1.1 (2-5-2024)

Added modules and modifications

* Hard drive saved excel sheet of every exposure time for all channels and cycles
* Added tissue identification capabilities
* Added 11th layer to focus map object that handles information storage of if a tile contains tissue or now
* Added nan\_to\_num() function right after all image acquisitions. This removes nonsensical values immediately. (patch fix 2-26-2024).
* Sped up imaging time speed via elimination of time delay between z steps of 0.3 seconds and added 0.2 seconds time delay with tile movements (patch fix 2-26-2024)

### V1.2 (unreleased)

Added modules and modifications

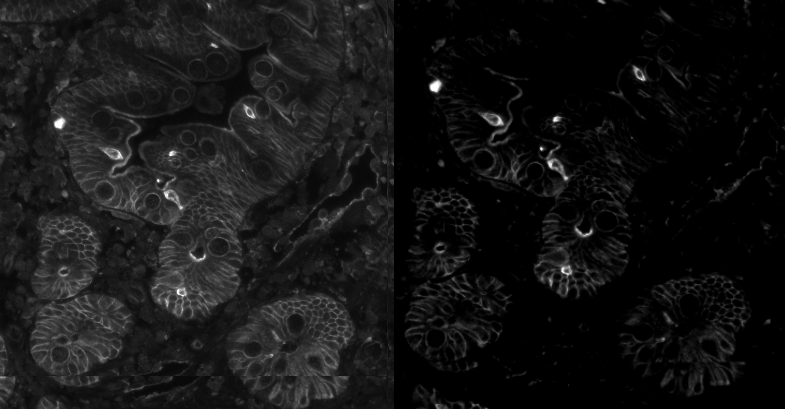
* Added 4 additional layers to focus map to store information related to how many images to average
* Added module function to capture the same spot a variable amount of times and give back the average of them

## Background Subtraction

Looking at line profile over a Na-K-ATPase stain line, we can see the background subtraction increased peak signal to bckgnd ratio from 1.32 to 2.25.

A screenshot of a computer

Description automatically generated



## Auto Focus

### 12-6-2023 change (V2.0)

Up until now I had been running an autofocus program which entailed taking images in a z stack at every tile and finding the max and centering my acquisition z stack focus map on that. It hit me that this can be simplified and hastened. During the stain phase, I can look through the previous cycles images and recenter the best focus plane for each tile and then update the focus map. This is done during a downtime activity so net time cost on the system is zero. [Presentation\Previous Cycle, Star-Dist Autofocus.pptx](Presentation/Previous%20Cycle,%20Star-Dist%20Autofocus.pptx)

## Auto Exposure

### V1.0

Very basic approach to autoexposure. Briefly, make histogram of pixel intensities for each tile. Chop off top 1% and determine the remaining highest intensity pixel. Make scale factor of this in comparison with target intensity and multiply original exposure time by this factor. Go through every tile in this manner and set the global exp time to be the lowest computed exposure time.

### V2.0 (unreleased)

Up until now I had been running an autofocus program which entailed taking images in a z stack at every tile and finding the max and centering my acquisition z stack focus map on that. It hit me that this can be simplified and hastened. During the stain phase, I can look through the previous cycles images and recenter the best focus plane for each tile and then update the focus map. This is done during a downtime activity so net time cost on the system is zero. [Presentation\Auto Exposure V2.pptx](Presentation/Auto%20Exposure%20V2.pptx)

## Auto Tissue Identification

### V1.0 (2-5-2024)

Takes in StarDist segmentated nucleus binary image and dilates with 70 pixel diameter disk to fill in spaces between nuclei. Uses size filter to eliminate many smaller region that aren’t part of the main tissue. Marks 1 if any tissue exists, 0 if none exists in tile. Applies on a per tile basis.

* Converts nucleus segmentation to tissue via 70 pixel disk dilation
* Size filter per image
* Mark layer in focus map as 1 or 0
* Per tile basis

## Processing

### V1.0

***This version represents basic framework, but is not functioning****.* It has the following flow and features:

* Find in focus tiles for each channel and tile by taking center slice in Z stack
* Background subtract by using turbo stack to find displacement vector from cycle x bleached dapi to cycle x stained dapi and applying it to shift the stained channel image to align perfectly with its bleached counterpart and subbing the bleached version from it
* Use BaSiC to even out illumination via pybasic python library
* Assemble all processed images into a single stack and embed with proper metadata for McMicro to use
* Make all stained and bleached images placed into a larger image setup with pure stage coordinates (thus no stitching)
* Find in focus tiles for each channel and tile by highest brenner score in each stack with a derivative jump of 10 pixels (patch 2-14-2024)

### V1.1 (unreleased)

Following fixes and corrections were made

* Fully integrated auto tissue identification capabilities
* All functions now work due to poor pixel value correctors (ie. eliminated NaN and infinity values from images)

## Fluidics

### 2-22-2024 change (V1.1)

* Reversed order of establishing PID loop. Order change to initiate PID loop and then start remote loop. It was reversed before. Did this swap due to input from Elveflow.
* Added in fluidics logger that tracks all MUX and OB1 actions.
* Added in new variable to dictate if OB1 runs in pressure or flow control mode

# Issues

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Problem ID** | **Problem Description** | **Solution** | **Date** | **Person** | **Status** |
| I1 | Core capture got stuck in random spot. No good leads on why. Previously driver in mm had crashed. Not the case here. I restarted image acquisition and it had no issue. For reference, it had captured bleached cycle 1, stain/bleach for cycle 2 and stalled at stained for cycle 3. | Needed updates and a restart. Also must grab pic from core cap or if ask cap again without, will cause it to stall | 11/29/23 | MA | Resolved. |
| F1 | Pressure source loud and unable to sustain pressure >600mBar. | New pump fixed. | 11/28/23 | MA | Resolved |
| F2 | When asking to stop flow, a slow flow still exists. Its very minimal. Seems to make pbs hit the chamber in about 1 hour. | Do new calibration and ask for -3 flow as set point |  |  | Resolved |
| F3 | Only delivered cycle 1 stain in multiplex run. Nothing else was seemingly executed | Works sometimes. Contacted Elveflow to see if maybe instability. | 1/31/2024 | MA | Resolved |
| I2 | 6-2-24 multiplex Cycle 1, A647 was imaged and all pixel values were NaN. | No solution. Might be freak appearance. | 2/6/2024 | MA | Unknown |
| H1 | When heater on, bubbles form. | Most likely bubbles are due to evaporation. Might be too hot under the chamber | 2/20/2024 | MA | In process of getting different heater designed |
| C1 | Hard Drive Disappeared | Moved SATA cable to open SATA port on motherboard. | 2/23/2024 | MA | Resolved |
| P1 | BaSiC keeps failing and giving NaN on all pixels as output | Appears to be due to known NaN pixel values. Used nan\_to\_num function to correct | 2/23/2024 | MA | Resolved |

# Solution Testing

## F1

Pump is mega loud. Record 85db on phone and documents say should be 53db. Performance was not hindered until recently. As of now, it is only able to sustain 600mBar pressure. Solution is to do exchange with Elveflow for new unit as this one is clearly failing. This solution was correct.

## I1

Really hard to say here. It is beyond my abilities to resolve an issue with the core capture stalling out if it is due to architecture. Simply restarting the program made it work though. A possible solution could be to make the capture function wait a bit. If it exceeds a certain time frame, it simply asks it to do it again. Python IDE gave no errors back in this case. It just didn’t proceed on in the code.

Other possible solutions could be to alternative code to capture image. No need to use mm core capture. I could use python package from photometric to capture as well. Its untested if both forms of capturing can be done at the same time or if mm called ‘dibs’ on the camera.

F2

OB1 showed 0 flow, but 20mBar pressure. I saw a drop form through the device every 5 minutes. That is slow, but significant. I flipped the pressure shutoff valve and it forced the pressure to 0 and the flow ceased. Code needs update to reflect this. \*Note this only semi worked. Having it fully calibrate itself and dictating the flow be -3 instead of 0 seemed to drive it to virtually stop flow.

## F3

This might result from an instability somewhere. Using the F2 fix, I found in a run that 2 of the 7 cycles has issues where either flow didn’t start or couldn’t stop. Same exact code. Why different results? I contacted Elveflow to talk to a software engineer about it. For this issue, I can have an issue with the fluidic system either not stopping or starting and I can fix it by simply stopping the script and restarting it. Here is a snippet of what happened when I tried a tactic of if it failed, reset the ob1 and try again with the order. Finally resolved this issue. It stemmed from both the OB1 and Pressure source being plugged into the same power strip. Rarely when the pressure source kicked on, it causes a drop in current to the OB1 and resulted in instabilities. Oddly enough only the PID loop is affected. The pressure layer of the software remained stable.

## H1

Tried PBS in chamber and no water around it. Plugged in heater for 2 hours and not a single bubble formed. Interesting. Not sure why we got some the other day and none here.

A screenshot of a computer program

Description automatically generated

Made data logger, but has not revealed anything significant yet. I was informed that usb hubs and shared power circuits between pumps and the OB1 are known to cause instability issues. I do not have a joint circuit, but I all of my devices are run from the same USB controller on the computer. Maybe that is like the external USB hub? Hard to say. The software engineer from Elveflow has been pretty unresponsive though.

I heard back from Elveflow and they and they had me swap two functions. I had start remote loop and then start PID. It needs to be reversed. I tested and it worked. Also, I tried out the data logger with the fluidics for a real run and it failed twice. Originally, I felt that the read off was accurate, but now I am not sure. On one failure, it said my flow was -3.03 instead of 500. However, it emptied the vial, which points to that it actually did the command. In addition, both of the times that it failed, the flow read off gave back a value of -3.03.

As of now (20-2-24) I believe the following to be true:

1. The PID loop is what messes up
2. The PID doesn’t mess up in vacuum and other microscope actions must be taking place too.

I had both the OB1 and pump on same power strip. Might be the cause of the issue. I separated them and am testing on 2/21/2024 to see if any issues arise.

## P1

I did some testing and it appears that some images have values that are not defined (its like an empty cell). Here is an example with my cursor over the undefined pixel. Its very odd. As of now, I have no explanation for why the pixel isn’t populated. Its an easy correction from a code side as I just need to add in the function np.nan\_to\_num(). It fixes, but doesn’t explain. I have confirmed that if I run this function and use BaSiC on an image that originally failed, it now works.

A screenshot of a computer

Description automatically generated

# Versioning

## V1.0

debuted 12-12-2024 in the Elveflow branch. First fully successful acquisition code. Currently resides in legacy in the 12-12-23 branch. Contained the following modules:

* V2.0 Auto Focus
* V1.0 Auto Exposure
* V1.0 Fluidics
* V1.0 Processing
* V1.0 Acquisition Engine

## V1.1

Debuted 2-5-2024 in the Elveflow branch. Current version in Elveflow branch and verified to work in testing. Contained the following modules:

* V2.0 Auto Focus
* V1.0 Auto Exposure
* V1.1 Fluidics
* V1.0 Processing
* V1.1 Acquisition Engine
* V1.0 Tissue Region Identification

## V1.2

Unreleased. Currently on 2-2-24 auto exposure branch. Contains the following modules:

* V2.0 Auto Focus
* V2.0 Auto Exposure
* V1.1 Fluidics
* V1.1 Processing
* V1.2 Acquisition Engine
* V1.0 Tissue Region Identification