# Week of 9/25 Deliverables

CLARITY

# NEW NAME

- COBALT: CLARITY-Optimized Brain AnaLysis Tools

# Last week’s goals

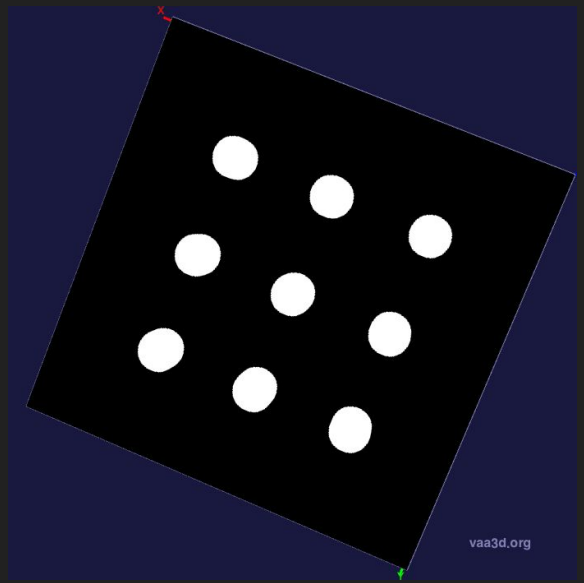
- Complete Project proposal
- Complete sprint timeline
- Profile ndreg code
- Create simulated image generation tool
- Explore performance of watershed/blob detection qualitatively
  - ✓ Simple Blob Detector
  - ✓ Distance Transform for Cell Overlaps

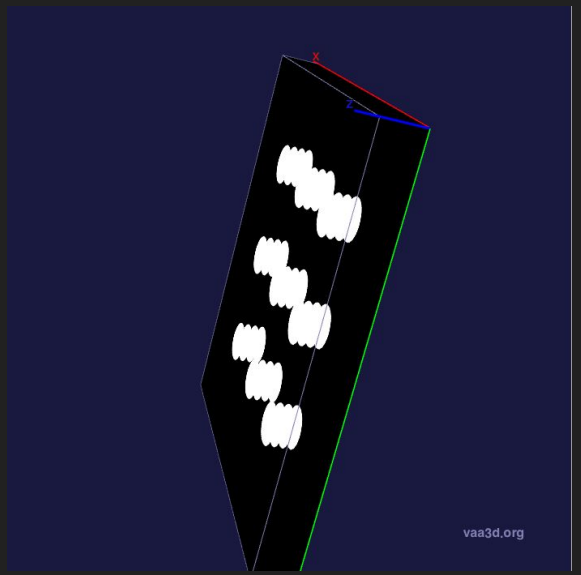
# Sprint Timeline

| Sprint | Date Due | Requirements: Portion of manuscript (and package) corresponding to... |
|---|---|---|
| Sprint 1: Registration and Cell Detection | 10/18 | • Obtain/Label up to 10 subvolumes of interest with manual annotations for cell detection<br>• Tool for generating simulated validation data<br>• Implement & quantitatively compare 2 unsupervised cell detection methods<br>• Quantitative evaluation of 25, 50 um registration on 4 control brains with fiducials |
| Sprint 2: | 12/8 | • Well-documented cell detection python package (w/ 3 algorithms for cell detection)<br>• Region-based analysis in package<br>• Voxel-based statistics |
| Sprint 3: | 3/5 | • Obtain/Label up to 10 subvolumes of interest with manual annotations for tractography<br>• Tool for generating simulated validation data for tractography<br>• Implement & quantitatively compare 2 unsupervised tractography methods<br>• Create a web service that runs cell detection on 1 TB volumes |
| Spring 4: | 5/16 | • Expand front-end web service to run tractography on 1 TB volume of data<br>• Inlcude tractography in cell detection python package with comprehensive documentation and 2 tractography algorithms |

# Cell counting image generation tool

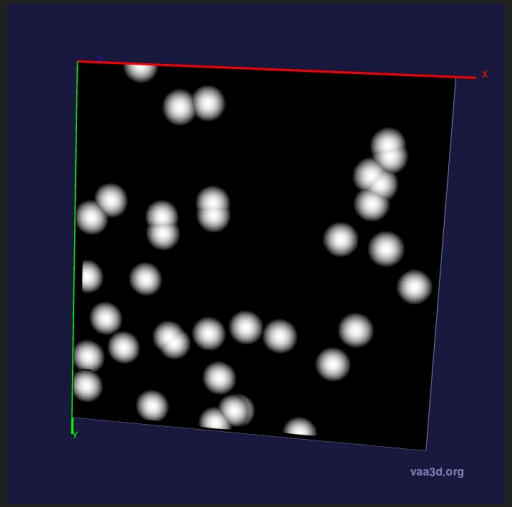
- Inspired by Ailey-dev, created a Clarity-like image generation tool.
- Tool is very basic for now. Able to create ellipsoidal like objects with different properties like size, locations, colors, fade, intensity.
- It is very customizable and anyone on the team can use it easily.



# Tool cont.

- We plan on using these images to evaluate and validate our algorithms. We can control how many cells are in each image, the overlapping of cells, their distribution, their shapes, the sizes of the volumes, etc (in future want to make the cells irregular shapes)
- if we go to deep learning then suing generated training data like this is invaluable.





- Code: https://github.com/NeuroDataDesign/clarity-f17s18/blob/master/source/jyim6/util/ImageGenerator.py
- Notebook (just my workspace): https://github.com/NeuroDataDesign/clarity-f17s18/blob/master/source/jyim6/Simulated%20data.ipynb

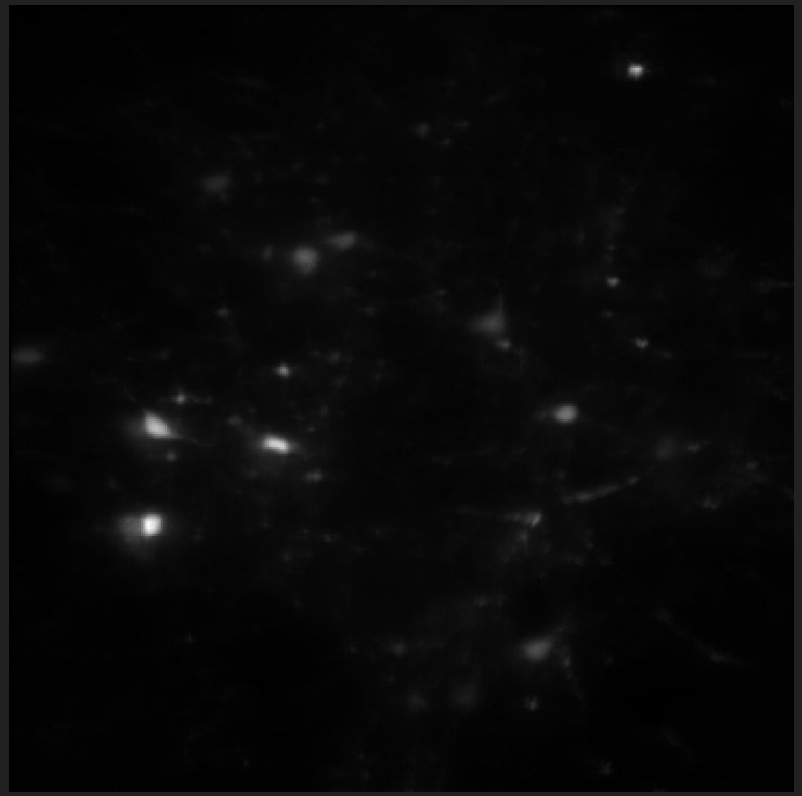
# Overview of Srivathsa's work

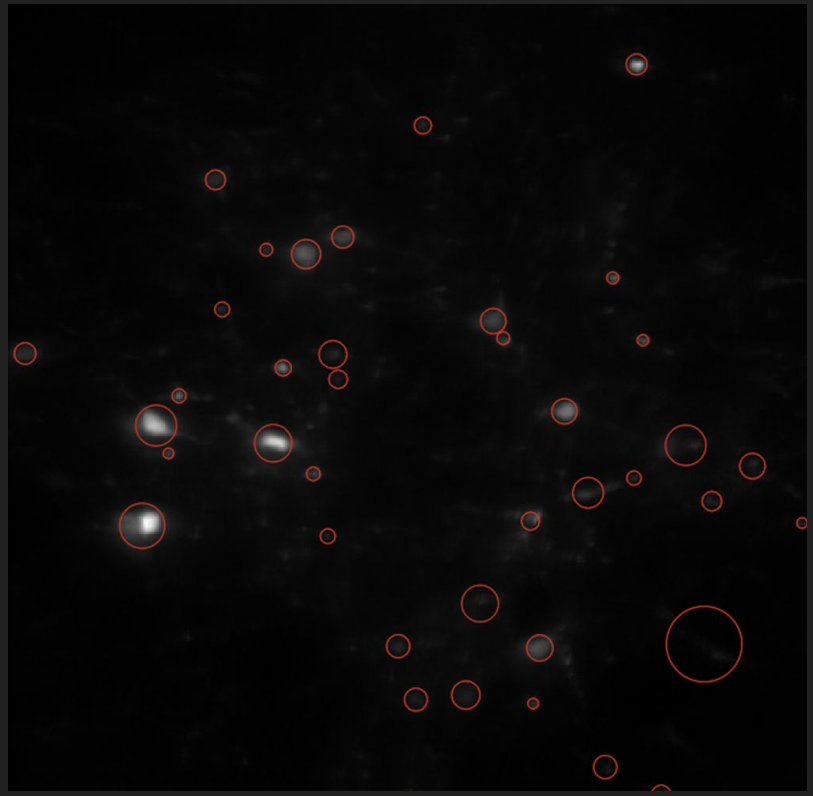
- Based on last week's results, **Watershed Algorithm** showed good promise in cell detection.
- Implemented basic 2D cell detection script using OpenCV's `SimpleBlobDetector`
- Implemented *Distance Transform* based `SimpleBlobDetector` to tackle the problem of overlapping cells

# Simple Blob Detector

- Output of OpenCV's SimpleBlobDetector which is implemented based on
  - Watershed algorithm
  - OTSU's binarization threshold
  - Scale Invariant Feature Transform (SIFT)
- False positives are very less
- Can get very good results if combined with 3D reconstruction algorithms

# Simple Blob Detector

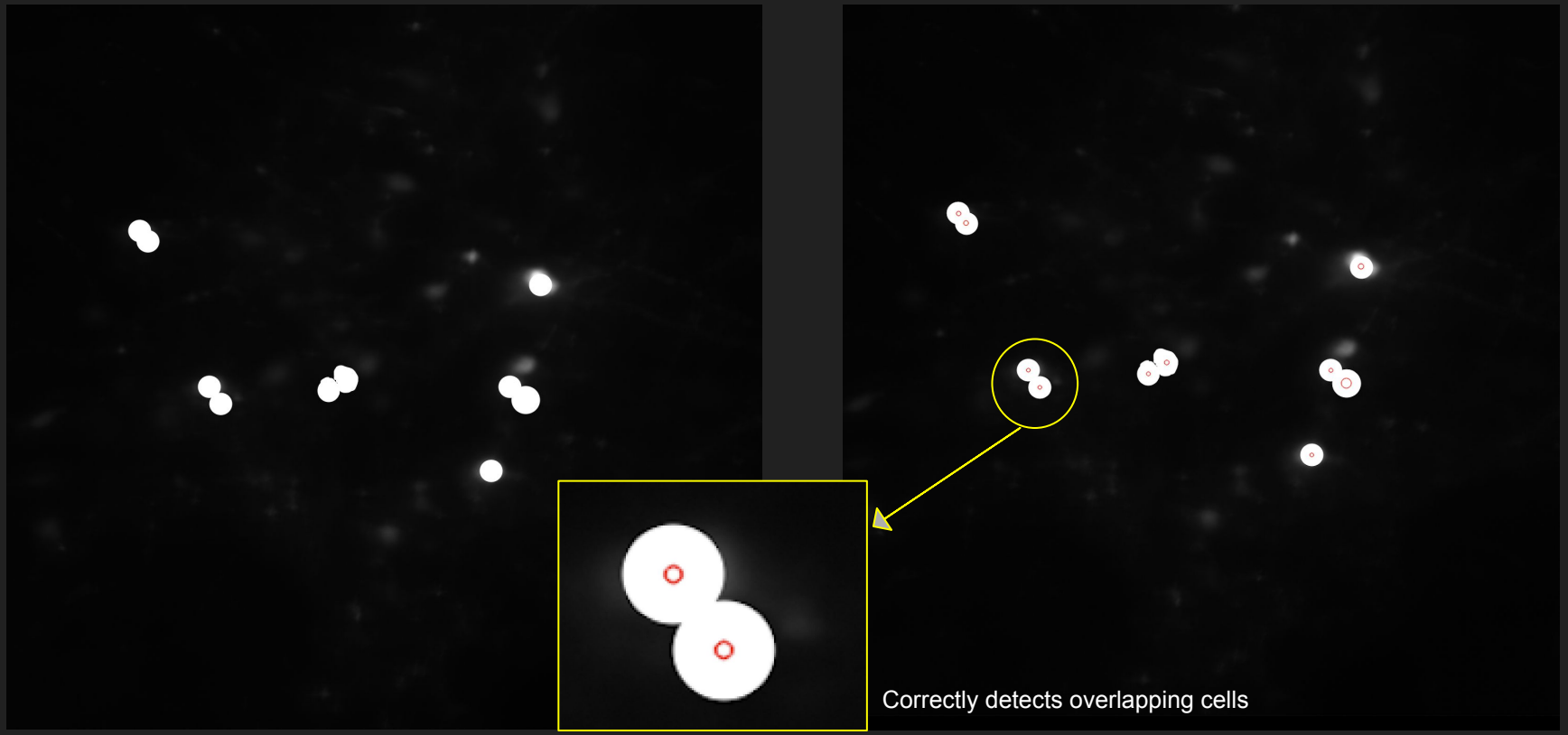




# Distance transform for overlapping cells

- **Morphological opening** is used to create sure background area and **distance transform** is performed for sure foreground area.
- This removes ambiguity between overlapping cells and identifies them individually

# Distance transform for overlapping cells



# NDREG Profiling - 50 um ARA registration of s3617

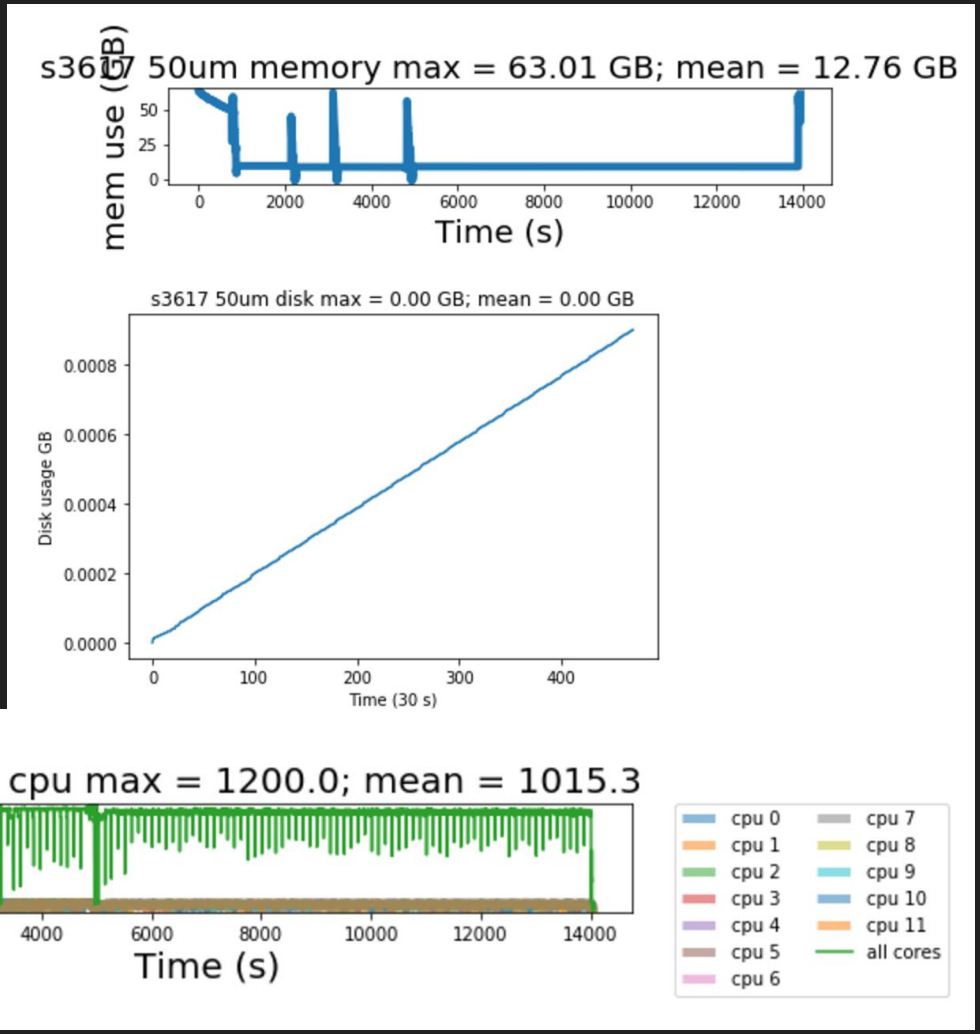
- notebook of profiling
- registration code
- Wasn't able to profile lddmm code
  - Will try running it on machine with more RAM

```
Step 0: alpha=0.2, beta=0.05, scale=1.0
Error: Metamorphosis did not terminate normally.

itk::MemoryAllocationError (0x7fc768000940)
Location: "TElement* itk::ImportImageContainer<TElementIdentifier, TElement>::AllocateElements(itk::ImportImageContainer<TElementIde
ntifier, TElement>::ElementIdentifier, bool) const [with TElementIdentifier = long unsigned int; TElement = itk::Vector<double, 3u>;
 itk::ImportImageContainer<TElementIdentifier, TElement>::ElementIdentifier = long unsigned int]"
File: /usr/include/ITK-4.9/itkImportImageContainer.hxx
Line: 199
Description: Failed to allocate memory for image.


Traceback (most recent call last):
  File "reg_s3617_50um.py", line 201, in <module>
    scaleList = 1.0, useMI=True, iterations=100, verbose=True)
  File "/root/.local/lib/python2.7/site-packages/ndreg-0.0.1-py2.7.egg/ndreg/ndreg.py", line 1329, in imgMetamorphosisComposite
    outDirPath=stepDirPath)
  File "/root/.local/lib/python2.7/site-packages/ndreg-0.0.1-py2.7.egg/ndreg/ndreg.py", line 1254, in imgMetamorphosis
    (returnValue, logText) = run(command, verbose=verbose)
  File "/root/.local/lib/python2.7/site-packages/ndreg-0.0.1-py2.7.egg/ndreg/ndreg.py", line 85, in run
    if checkReturnValue and (returnValue != 0): raise Exception(outText)
```

# NDREG Profiling - graphs



# Next week

- Send email to Li Ye to request manually labelled COLM data
- Create 10 simulation datasets
- Evaluate 4 unsupervised cell detection algorithms (LoG, DoG, DoH, watershed) on 10 different simulated datasets
  - Compute true-positive, false-positive
  - Compute F-measure (precision and recall)
- Write detailed pseudocode for 3 unsupervised methods from literature (see algorithms.md)
- Quantitatively evaluate Thy1eYFP_Control_9 50um registration to ARA using manual fiducials
  - Compute mean-squared error