Supervised Machine Learning for Biomolecular Condensate Quantification and Measurement

Rodney Lafuente Mercado
lafuentemercado@college.harvard.edu

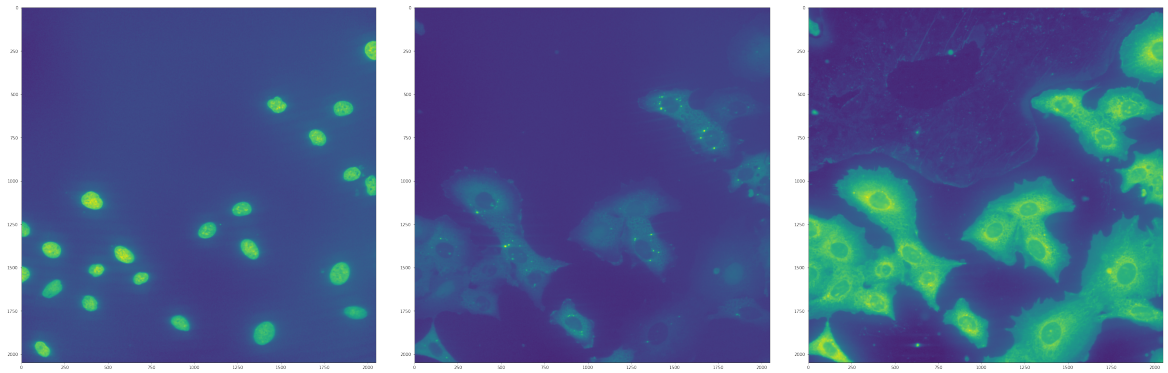
May 4, 2022

Abstract

This report details the background, approach, and process of applying supervised machine learning methods for identification of biomolecular condensates in SARS-CoV-2 nuclecaspids. The report covers not only the feature and model selection of the process but also the stuff regarding splitting and segmenting and filtering as well.

1 Background

The motivation behind this project is to quantify the effect of different compounds on condensate formation in SARS-CoV-2 nucleocaspids. This is part of a larger study in what things affect what in what. Current methods such as software like MetaXpress are effective but there is suspicion that machine learning may be way better at the identification. The method outlined in this paper is inspired by Ilatisk, albeit with different methods to handle false positives. Images analyzed in this project were of these nucleocaspids under three different wavelengths. These three wavelengths are visualized in Figure 1 for an example image (KRD-211028-p5_G24_s4).



(a) Wavelength 1 (b) Wavelength 2 (c) Wavelength 3

Figure 1: Example of Well Site Wavelengths

the wavelengths displayed each play a different purpose as is illustrated below. This report will

split the process into three sections, namely the approach to cell segmentation, feature selection, and model selection.

2 Cell Segmentation

Machine learning methods in the approach outlined in this report work only to classify pixels as belonging to a biomolecular condensate or not. An additional component in the process of identifying and measuring condensates is to identify areas of interest in an image, namely, differing regions that belong to cells. Once these regions of interest are identified, condensates located in these regions, and only in these regions, can be quantified.

2.1 Approach

Cell segmentation was accomplished using watershed segmentation, an algorithm that takes as input binary images that it treats as topological maps such that they are flooded in order to find the proper separation between regions. This method, as well as all other methods outlined in this report, are implemented using the ScikitImage and ScikitLearn Python libraries. For identifying separate regions, a mask identifying nuclei was used. For identifying regions containing cytoplasm, a mask identifying cytoplasm was used.

The masking process, an example of which can be seen below, involves the computation of a certain threshold that must be high enough such that only pixels of a desirable intensity are retained. The method for nuclei mask creation was to use Otsu's method of thresholding. This method aims to minimize the variance between . The two thresholding methods were chosen by applying a variety of different thresholding mechanisms to the images and picking the ones best differentiating between cells in the end.

2.2 Results

Figure 2 displays the full masking process, save for a step that removes cells from the mask that overlap with the an outlier mask

This method was accurate enough to meaningfully separate condensates by which cells they belonged to, but definitely had its drawbacks. While this method was not perfect for this perfect, it is difficult to imagine a better method given the difficulty of segmenting cells by hand; boundaries between cells are, more often than not, not clear in the images in which they are presented.

3 Feature Selection

Classification of pixels into condensate and non-condensate regions requires not only a labeled data set of pixels but also a way of determining which properties of those pixels are to be used for classification purposes. For this model, we ran two different types of features: Gaussian Blurring, for determining intensity at the pixel and in areas surrounding the pixel, as well as eigenvalues of the Hessian matrix

3.1 Approach

Supervised machine learning requires not only a labeled data set but also a method of extrapolating information on that data prior to discerning between different classes for each of the data points.

3.2 Results

Figure 3 shows a zoomed-in portion of an example well site.

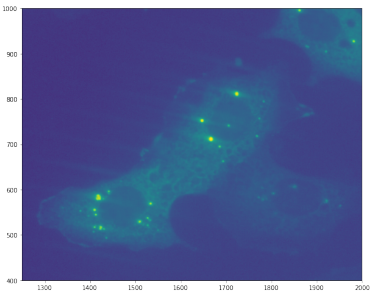
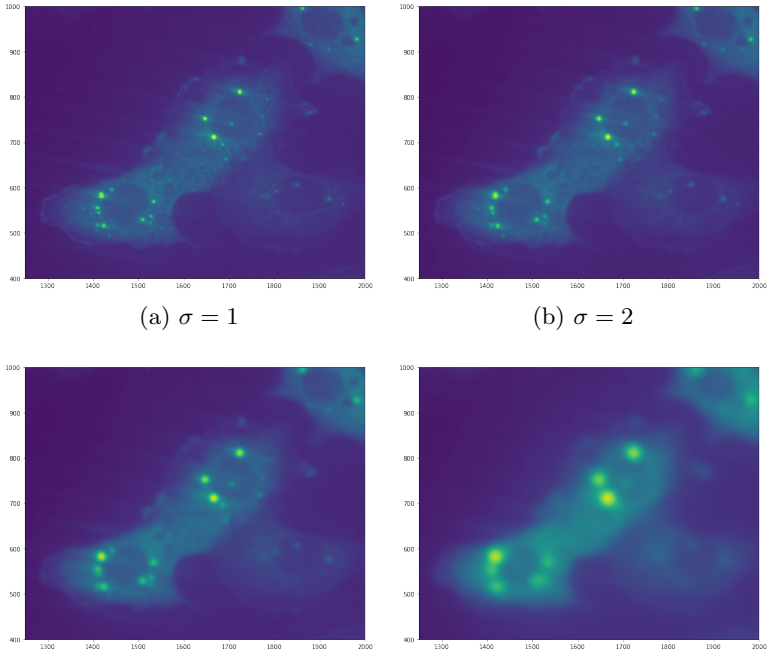


Figure 3: Zoomed Portion of Well Site

Figure 4 shows the outcome of applying several blurring filters to an example well site.

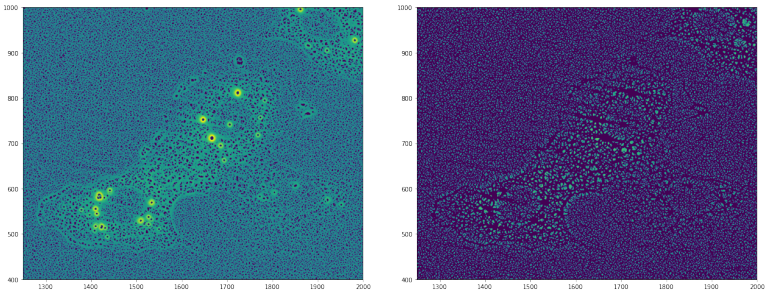


(a) $\sigma = 1$ (b) $\sigma = 2$

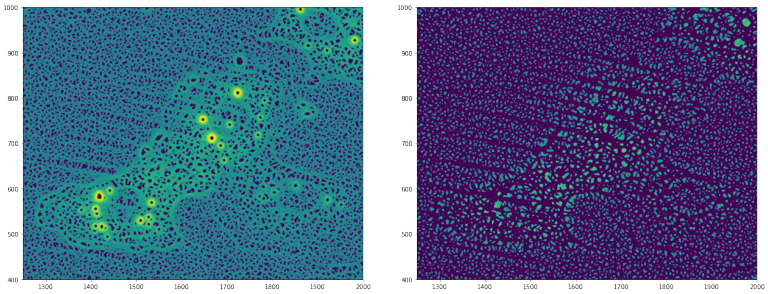
(c) $\sigma = 4$ (d) $\sigma = 8$

Figure 4: Gaussian Blurring Filters

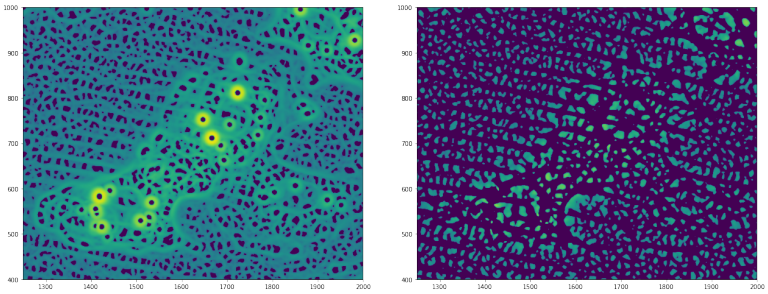
Figure 5 shows the outcome of applying several blurring filters to an example well site.



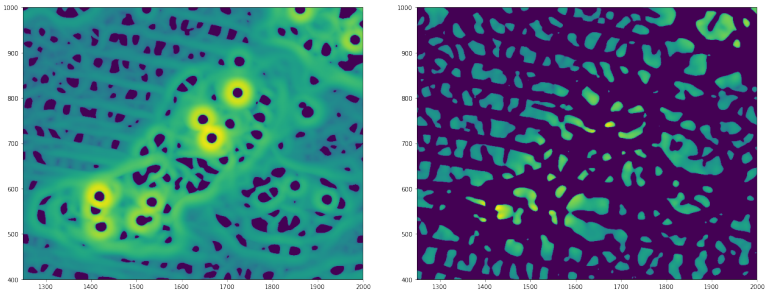
(a) $\sigma = 1$ (b) $\sigma = 2$



(c) $\sigma = 1$ (d) $\sigma = 2$



(e) $\sigma = 1$ (f) $\sigma = 2$



(g) $\sigma = 1$ (h) $\sigma = 2$

Figure 5: Texture Filters

4 Model Selection

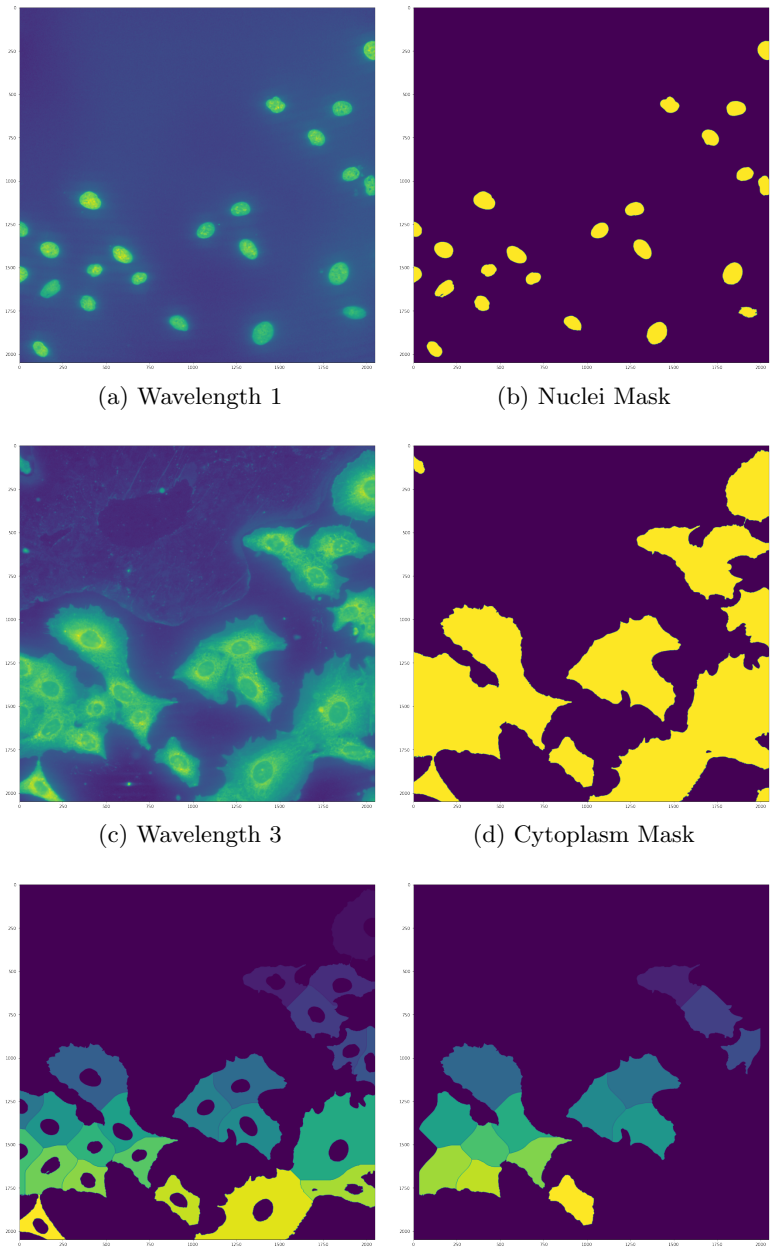
Ilastik, the software that implements similar machine learning techniques for pixel classification, uses random forest classifiers. We opted for this same model class for our dataset given that RFCs are effective in modeling distinctions in data points from data sets of limited length.

4.1 Approach

What did you do? When relevant, provide mathematical descriptions or pseudocode. Credit will be given for:

4.2 Results

This section should report on the following questions:



(a) Wavelength 1

(b) Nuclei Mask

(c) Wavelength 3

(d) Cytoplasm Mask

(e) Segmented Cells (Nuclei for clarity)

(f) Resulting Mask (Border and Outlier Cells Removed)

Figure 2: Full Masking Process