

# Size and Density Independent Cryo-EM Particle Selection via Positive-Unlabeled Learning

Yesh Doctor

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	Cryo-EM as a Method for 3D Structure Resolution . . . . .	2
1.2	3D Structure Reconstruction via Precise Particle Selection . . . . .	2
<b>2</b>	<b>Methods and Results</b>	<b>3</b>
2.1	Micrograph Preprocessing Pipeline . . . . .	3
2.1.1	Image Selection . . . . .	3
2.1.2	Contrast Enhancement . . . . .	3
2.1.3	Thresholding and Contour Extraction . . . . .	3
2.1.4	$K$ -means Clustering To Remove Small and Large Contour Artifacts . . . . .	3
2.1.5	Contour Pruning via Histogram Binning . . . . .	4
2.1.6	Futher Contour Pruning via Bounding Particle Size . . . . .	4
2.1.7	Size and Density Estimation . . . . .	4
2.2	Evaluation on EMPIAR-10216 . . . . .	4
2.3	PreProcessing Pipeline Works Better on Small, Densely Distributed Particles . . . . .	5
2.4	Model Trained on EMPAIR-10216 Generalizes Poorly to Similar Datasets . . . . .	5
<b>3</b>	<b>Discussion</b>	<b>5</b>
3.1	Preprocessing Pipleine Estimates Particle Size and Density "Well Enough" for Topaz . . . . .	6
3.2	Methods to Improve Particle Size and Density Estimation . . . . .	6
3.3	Methods to Improve Topaz Particle Extraction . . . . .	7
3.4	Conclusion . . . . .	7
<b>4</b>	<b>References</b>	<b>8</b>
<b>5</b>	<b>Figures</b>	<b>9</b>

# 1 Introduction

## 1.1 Cryo-EM as a Method for 3D Structure Resolution

Cryo electron microscopy (Cryo-EM) is a powerful technique for 3D protein structure resolution. This technique involves freezing aqueous proteins to liquid-nitrogen temperatures [1] and using a tunneling electron microscope (TEM) to image each protein ("particle") in multiple different confirmations / angles. [2] These images are called micrographs, and by "picking" (drawing a bounding circle around) particles from micrograph images (typically called a micrograph stack), a 3D conformation of the underlying protein structure can be reconstructed. [3,4] Cryo-EM is versatile in that it can capture images of particles ranging from Mega to hundreds of Dalton's at single-digit angstrom resolution. [5] Compared to other methods of 3D protein structure resolution (X-ray crystallography, NMR spectroscopy), Cryo-EM requires a far lower sample concentration and is capable of imaging a wider range of particle sizes. [6] The generated micrographs are typically noisy images visualized in gray scale with darker regions representing particles and lighter regions being the background ice-film. Samples are loaded into a carbon-mesh grid, so carbon and ice are typical high-frequency artifacts present in Cryo-EM imaging that decrease the resolution at which proteins can be reconstructed. [7]

## 1.2 3D Structure Reconstruction via Precise Particle Selection

Protein structure reconstruction requires selection of suitable particles from a micrograph stack, and is ultimately determined by the number of particles successfully selected. [8] Classically, this has been performed by manual methods where the user would have to select particles for reconstruction. Several image-processing based accelerations to this task have been implemented. One popular approach is cross-correlating each micrograph with a Gaussian disk, and selecting particles by non-maximum suppression. [9] However these methods are prone to under selecting particles, and require some extent of user data inspection / pruning. Furthermore, these methods assume knowledge of the approximate size of the particles, often leading to false-positives and struggles with identifying irregularly shaped particles. [10] Recently, there have been advances borrowing from the computer vision community to create machine-learning based algorithms to select particles. [10,11] Topaz is a convolutional neural network (CNN) based particle picker that leverages positive-unlabeled (PU) learning to select particles with low false-positive rates. A key benefit of PU learning is it requires a small number of ground-truth labels to train the CNN model. [12] Topaz is capable of selecting unusually shaped proteins (e.g. non-globular) with high fidelity while being trained on sparsely labeled data, making it a robust particle selection tool compared to classical (Difference of Gaussian (DoG)) methods. Topaz also selects more particles than DoG methods, resulting in higher resolution of the reconstructed protein. However, Topaz requires a priori knowledge of both particle size (radius) and particle density within each micrograph, which are often unknown values. Henceforth, particle density will refer to the number of particles within a micrograph, not the true density of each particle. To create a robust particle selection pipeline, we seek to leverage classical image processing techniques to develop a rough estimate of particle size and density, and couple these estimates with Topaz to develop an end-to-end particle picking pipeline that does not require manual input or visual inspection. To the best of our knowledge, this would be the first particle selection pipeline that requires no user input of estimated particle size or density.

## 2 Methods and Results

### 2.1 Micrograph Preprocessing Pipeline

Here we describe a micrograph preprocessing pipeline which outputs a size and density estimation of the particles in that micrograph stack. The general approach we use is to draw contours around each particle, using the most-likely contours as an indication for size (assuming homogeneity) and the number of contours as an indication for density.

#### 2.1.1 Image Selection

For each micrograph stack where we aim to extract particles, micrographs are rescaled to 256 x 256 images. If the original micrographs are not of the same aspect ratio, a 256 x 256 version is created through interpolation using OpenCVs interpolation algorithm. [13] For all micrographs in a stack, we measure a defocusing parameter and Gaussian disk cross correlation parameter that in tandem indicate images with the highest contrast. We first bin images into 10 categories based on the cross correlation values. Using only images from the bin contain the highest cross correlation we select the ten images that have the highest focusing value. These images are used for initial density and size estimation.

#### 2.1.2 Contrast Enhancement

Images are then blurred using a 7 x 7 Gaussian blur kernel with a standard deviation of 0.1. Using Gaussian blurring helps to isolate high contrast features (true particles) from the background by homogenizing the background pixel intensity. True particles tend to have sharp pixel differences from the background. Convolution with an edge detecting filter was also attempted however a large performance increase was not seen and therefore is not used in the final pipeline. Blurring with 3 x 3 and 5 x 5 kernels lead to similar performance as the 7 x 7 kernel. We also artificially increase image contrast by using 10x contrast stretching (normalized to the background).

#### 2.1.3 Thresholding and Contour Extraction

Following blurring, images are thresholded using Otsu's thresholding creating a binary map of regions containing high and low spatial frequency. From this map we extract contours using OpenCV's find contour algorithm. [13] (simple chain approximation method). We further narrow the number of contours by only selecting contours that are in the outer most contour hierarchy. This procedure helps to prevent double counting particles such as Apoferritin which have holes in the center of them. Selecting the outer most contour in the hierarchy will only highlight edges surrounding particles and not edges within particles themselves.

#### 2.1.4 K-means Clustering To Remove Small and Large Contour Artifacts

One observation from initial testing is that this method tends to select a large number of noisy contours that are smaller than the particles that are being selected. These contours are difficult to remove programmatically through Fourier filtering due to different unknown filter parameters being required for every particle stack. To combat this issue, we utilize one dimensional  $K$ -means clustering. We cluster contours based on contour area using  $K = 1$  to  $K = 100$ . We then calculate the average cluster centroid deviation for each cluster and divide by the number of clusters for  $K = 1$  to  $K = 100$ . This tends to produce an exponential decay curve ( $K$  on the x-axis, average cluster centroid deviation on y-axis) which we can fit using a simple model of the form  $Ae^{-kt} + B$ . We use scipi-optimize to achieve this fit. We then calculate the gradient of the fitting and determine the point at which the gradient changes between two  $K$  values by an amount  $\leq \epsilon = 1$ .  $\epsilon$  is a hyper

parameter that determines the extent to which  $K$ -means represents the true distribution of contour areas. We choose  $\epsilon = 1$  reduce over fitting.

### 2.1.5 Contour Pruning via Histogram Binning

Using the  $K$ -means model that achieved  $\epsilon = 1$ , we create a histogram where each bin represents a cluster (having value equal to the cluster centroid) and the count represents the number of contours in that cluster. This histogram tends to be left-skewed as this technique tends to pick many contours that are small and a few contours that are large. As will be discussed later, the pipeline tends to work better on data sets with smaller particles size and denser particle distribution. From this histogram we create lower and upper bounds on particle size. The first bin in the histogram tends to contain contours that have been isolated from background noise rather than true particles. The last bin tends to contain contours that enclose multiple particles as one. We initially prune contours by removing all contours in the first and last bins.

### 2.1.6 Futher Contour Pruning via Bounding Particle Size

We further narrow contours by creating lower and upper bounds on particle size. This is done by choosing only particles with areas greater than or equal to the the second bin's centroid minus twice the average cluster centroid deviation of that bin (lower bound), and area less than or equal to twice the second-to-last cluster centroid plus twice the average cluster deviation of that bin (upper bound). We find among the data sets that we have tested, this method tends to encompass most of the true particles while eliminating many false positives due to the background.

### 2.1.7 Size and Density Estimation

To estimate particle size we assume particles are roughly globular in shape. Using the cluster with the highest peak in the histogram (after pruning) we divide the cluster centroid by  $\pi$  and take the square root to get an approximate radius. We then multiply this radius by the scaling factor used to rescale images down to 256 x 256 to get the original particle size. To estimate particle density we divide the integrated area of the histogram by 10 (the number of images used). This allows us to find initial estimation of both particle size and density, with the ability to use these parameters as inputs to the Topaz particle selection model.

## 2.2 Evaluation on EMPIAR-10216

This particle estimation pipeline was initially tested on the publically available micrograph stack Electron Microscopy Public Image Archive (EMPIAR)-10216 (mouse heavy-chain apoferritin at 1.62 angstrom). (Figure 1, 2) We first trained Topaz for 10 epochs on 10216 using a 80/20 train/test split and five-fold cross validation. Models were seeded with initial particle density estimations of 200, 400, 600, 800, and 1000 particles. Area under the precision/recall curve (AUPRC) was used as the metric for model performance. Use initial particle density estimations of 400 through 1,000, did not lead to improved performance as training progressed from the epoch 1 to 10. However, using and initial estimation of 400 particles per image led to higher final AUPRC than using 1000 particles per image. This indicated that Topaz is indeed sensitive to particle density estimation. The data set trained with initial estimation of 200 particles per lead to the best performance, and performance improved as training progressed. This indicates that the true particle density of 10216 is approximately 200 particles per micrograph. Additionally 10216 contains protein micrographs of Apoferritin which is a protein containing a hole in the center, appearing as a light spot in the center of the particle; a challenging for detection. Using

N	Loss	Precision	True Positive Rate	False Positive Rate	AUPRC
124	0.0201	0.657	0.671	0.005	0.818
200	0.0233	0.704	0.703	0.004	0.815
400	0.1262	0.349	0.946	0.038	0.686
600	0.3157	0.249	0.976	0.048	0.617
800	0.4054	0.213	0.969	0.058	0.548
1000	0.82391	0.139	0.986	0.099	0.500

Table 1: Model (Epoch 10) Performance Metrics (N = Density Estimate)

the established pipeline, we estimated and initial particle density of 134 particles per image. When Topaz was trained on 10216 using 134 as initial park odyssey estimation, it outperformed The model trained using 200 as the initial particle density estimation. This validated to us both numerically and visually that the establish pipeline (when images are of high contrast) is able to automatically select particles with high fidelity and without manual input.

### 2.3 PreProcessing Pipeline Works Better on Small, Densely Distributed Particles

We tested the preprocessing pipeline on 5 data sets containing particles of different size, morphology, contrast, and artifact contamination. (Figure 3) We found that the preprocessing pipeline best estimated particle density on data from SD2 and BT1, which are smaller, more dense particles. When compared to VP2, BT1 and SD2 both have more of the particles selected with fewer background elements selected. In VP2 (the largest particle of the ones tested as it is a virus), the pipleine appears to be confused by particles closer to the carbon hole edge (such as in image 3), indicating that choosing particle images with minimal artifacts is important for density estimation. Furthermore, low-contrast images such as Apo2 and ice-contaminated images such as GDH also prove to be an issue for the estimation pipeline. However, the pipeline does seem to do a good job of determining an appropriate image size; Figure 3 (A,B,E) all have contours drawn around the particles, and in isolating exclusively the highest peak (data not shown) contours are drawn exclusively around some of the true particles and not in the background.

### 2.4 Model Trained on EMPAIR-10216 Generalizes Poorly to Similar Datasets

In Figure 4, we extract particle locations in SD2 and BT1 using the model trained on 10216 as our preprocessing pipeline indicated that these stacks have similar particle density and shape. In Figure 4 (A), particles appear to be psuedo-randomly selected as most of the blue circles indicating particle picks do not coincide with the actual particles. A similar story is seen in Figure 4 (B). In Figure 4 (C), BT1 images are rescaled to the same dimensions as the original 10216 images, which lead to slightly better performance. However, there are still quite a few particle picks that are not covering any particles (False positives) which could ultimately hurt image reconstruction.

## 3 Discussion

We have implemented a micrograph pre-processing pipeline that estimates particle density and size and uses these parameters in a machine learning model to select particles without manual input or pruning. The pipeline leverages high contrast features of the micrographs and selects for particles based on the assumption that they are roughly of similar size and morphology. We propose a novel method to particle selection wherein the process of determining particle size and density is automated, and a Topaz based neural network

is able to select particles by pre-training the model on a well labeled data set of similar particle size and density. This allows for an end to end pipeline that does not require user input to reconstruct protein 3-D structure.

### 3.1 Preprocessing Pipeline Estimates Particle Size and Density "Well Enough" for Topaz

The pre-processing pipeline we use appears to be effective at selecting particles when they input images are of high contrast. By selecting for images with high defocusing and cross correlation parameters we are able to generate an estimation of the particle density for all images in the data set using only a few selected images. This allows for flexibility in the number of micrographs; only a few number of high contrast micrographs are required to estimate critical parameters. This, coupled with the benefits of positive unable learning results in a large number of particles being selected without requiring any user input for ground truth value. We initially use Gaussian blurring as a means to homogenize the background and make particles stand out. In experiments where Gaussian blurring was not used (data not shown, thresholding resulted in a large number of contours being highlighted that were part of the background and not true particles. Furthermore we effectively prune the contours selected through one dimensional  $K$ -means clustering. From the experiments done with EMPIAR-10216, model performance was significantly improved when an initial density estimation of 134 particles was used compared to a naïve estimation without pruning of 314 particles. Additionally, from visual inspection using post-thresholding pruning eliminated small contours that was selected in the background. Implementing the restriction that contours selected have to be in the outer most hierarchy level improved performance on oddly shaped particles such as Apoferritin. When contours of any hierarchy level were used previously (data not shown) we found that contours were being drawn both outside and inside particles with holes in them. Imposing this restriction has increased accuracy with which the pre-processing pipeline estimates both size and density. Using the heuristic of the highest peak in the histogram being associated with the true particles seem to accurately pick true particles when the images used did not contain ice/carbon contamination. From visual inspection the radius predicted by this heuristic drew circles around particles that enclosed the entire particle, but did not enclose multiple particles in one circle. This allows for reconstruction without multiple particles being identified as one.

### 3.2 Methods to Improve Particle Size and Density Estimation

One key weakness of the pre-processing pipeline is that it performs poorly when there are contaminants such as ice or carbon in the initial image. The Bartesaghi Lab does have tools that are capable of removing such elements, and in future iterations of this pipeline those pre-processing tools will be prepended. Furthermore, the pipeline appears to struggle with selecting particles that are tightly grouped together. It tends to draw a contour around all particles that do not have a single defining boundary surrounding them. Hence in some images quite a few particles are removed from the density estimation as thier boundaries overlap. There are several methods we could potentially use to improve upon this. One method could be to normalize each histogram been by the centroid of the largest bin. Assuming that the largest bin contains single particles, this method would allow for an estimation of the number of particles included in each contour that is not in the central bin. This will account for multiple particles being selected as one since we have an estimation of the number of contours, and number of particles per contour. And additional method that could be used to improve this initial destiny estimation is to use more than 10 images. Although this increases the need for high contrast images, it would account for variability from image to image by law of large numbers. Finally a pairwise distance matrix from the center of all contours could be made. This matrix could then be

pruned for distances greater than the estimated particle radius (to remove particles that are on top of each other), and less than the dimensions of the image divided by the total estimated particle density times the estimated particle area. (To remove particles that are farther away than the expected largest distance between particles, assuming uniform particle distribution).

An additional limitation of the pre-processing model is the assumption that the first peak will contain particles that are actually in the background and not true particle labels. If there are particles that are very small, or very large, they will be removed by the first and last bin pruning. This could be adjusted through the parameter  $\epsilon$ , however this is a hyper parameter, and its current state, human observation is required to determine its value. A value of  $\epsilon = 1$  has worked for our smallest and largest data set tested, however this is not guaranteed that for all particles this value will be optimal.

### 3.3 Methods to Improve Topaz Particle Extraction

The Topaz CNN itself seems to overfit to the data set that is trained on, and it's not highly applicable to other data sets, even if the data sets are of a similar particle shape and density. After image scaling and optimal particle size and density is inputted into the extract function of Topaz, when a Topaz model trained on EMPIAR-10216 was used, the model poorly extrapolated to the SD2 data set. In some cases it was able to select particles, but there were many false positives (by eye test). One limitation of the study is that there are no known ground truth labels for any of the data sets used, so all inspection is strictly visual. Nevertheless there are methods that could be used to improve particle selection. Perhaps the most promising is to use a transfer learning approach, where Topaz is initially trained on data sets where which are well labeled (EMPIAR) and contain approximately the same number of particles and particle size as the micrograph stack of interest. We attempted to use models from earlier epochs of training (three, five, seven, nine). However the presumed overfitting issue still remained (data not shown), and we believe from experiments done with 10216, Topaz performance best when it is trained on the same micrographs that it is trying to extract from. The pre-processing pipeline can be used to select particles (as ground truth labels) by choosing the highest peak in the histogram and assuming side peaks are associated with true particles. Topaz has the benefit of leveraging positive unlabeled learning, which requires a few ground truth labels to be able to extract particles. A network that is pre-trained on a daily set of similar size and density could then be further trained on a few images from the stack of interest that have been labeled using the pre-processing pipeline. Then, particles from the entire stack could be extracted using the transfer learning trained model. Alternatively, the model could be trained every time on the micrograph stack of interest using a few positive labeled particles through the highest histogram peak method. We also observed that extracting on micrographs that are not the same dimension of micrographs that Topaz was trained on (512 x 512) took substantially longer than extracting on micrographs of the same dimension. This implies that cropping and interpolating the extracting images may lead to both faster and potentially more accurate particle location picks parentheses (this was observed on SD2 which has a native resolution of 1024 x 142, and extraction results were worse than when these images were rescaled to 512 x 512.)

### 3.4 Conclusion

Overall, we believe that we have established a robust pipeline for particle density and size estimation given micrographs of relatively high contrast and low contamination. We have found that these parameters can be coupled with the Topaz PU CNN to extract particles, and in conjunction the entire pipeline is able to select particles for 3-D reconstruction in a completely automated manner. To our knowledge this is the first reconstruction pipeline that does not require any user input, and while the pipeline is not applicable to

all data sets, we have shown promising results on similar data sets (SD2 and 10216) with the ability to effectively choose tricky particles (i.e. particles with holes in them) and extract their position with some accuracy. We anticipate that with some of the proposed improvements in this discussion, we will be able to enhance the pipelines performance, and be able to generate a completely automated way to select particles for Cryo-EM based protein structure resolution.

## 4 References

1. Dubochet, J., M. Adrian, J. J. Chang, J. C. Homo, J. Lepault, A. W. McDowall, and P. Schultz. 1988. "Cryo-Electron Microscopy of Vitrified Specimens." *Quarterly Reviews of Biophysics* 21 (2): 129–228.
2. Longchamp, Jean-Nicolas, Stephan Rauschenbach, Sabine Abb, Conrad Escher, Tatiana Latychevskaia, Klaus Kern, and Hans-Werner Fink. 2017. "Imaging Proteins at the Single-Molecule Level." *Proceedings of the National Academy of Sciences of the United States of America* 114 (7): 1474–79.
3. De Rosier, D. J., and A. Klug. 1968. "Reconstruction of Three Dimensional Structures from Electron Micrographs." *Nature* 217 (5124): 130–34.
4. Stewart, P. L., R. B. Cary, S. R. Peterson, and C. Y. Chiu. 2000. "Digitally Collected Cryo-Electron Micrographs for Single Particle Reconstruction." *Microscopy Research and Technique* 49 (3): 224–32.
5. Murata, Kazuyoshi, and Matthias Wolf. 2018. "Cryo-Electron Microscopy for Structural Analysis of Dynamic Biological Macromolecules." *Biochimica et Biophysica Acta, General Subjects* 1862 (2): 324–34.
6. Shoemaker, Susannah C., and Nozomi Ando. 2018. "X-Rays in the Cryo-Electron Microscopy Era: Structural Biology's Dynamic Future." *Biochemistry* 57 (3): 277–85.
7. Stewart, P. L., R. B. Cary, S. R. Peterson, and C. Y. Chiu. 2000. "Digitally Collected Cryo-Electron Micrographs for Single Particle Reconstruction." *Microscopy Research and Technique* 49 (3): 224–32.
8. Callaway, Ewen. 2020. "Revolutionary Cryo-EM Is Taking over Structural Biology." *Nature* 578 (7794): 201.
9. Penczek, Pawel A. 2020. "Reliable Cryo-EM Resolution Estimation with Modified Fourier Shell Correlation." *IUCrJ* 7 (Pt 6): 995–1008.
10. Al-Azzawi, Adil, Anes Ouadou, John J. Tanner, and Jianlin Cheng. 2019. "AutoCryoPicker: An Unsupervised Learning Approach for Fully Automated Single Particle Picking in Cryo-EM Images." *BMC Bioinformatics* 20 (1): 326.
11. Wagner, Thorsten, Felipe Merino, Markus Stabrin, Toshio Moriya, Claudia Antoni, Amir Apelbaum, Philine Hagel, et al. 2019. "SPHIRE-crYOLO Is a Fast and Accurate Fully Automated Particle Picker for Cryo-EM." *Communications Biology* 2 (June): 218.
12. Bepler, Tristan, Andrew Morin, Micah Rapp, Julia Brasch, Lawrence Shapiro, Alex J. Noble, and Bonnie Berger. 2019. "Positive-Unlabeled Convolutional Neural Networks for Particle Picking in Cryo-Electron Micrographs." *Nature Methods* 16 (11): 1153–60.

13. Kang, and Atul. 2018. “Image Interpolation Using OpenCV-Python.” November 15, 2018. <https://theailearner.com/2018/11/15/image-interpolation-using-opencv-python/>.

## 5 Figures

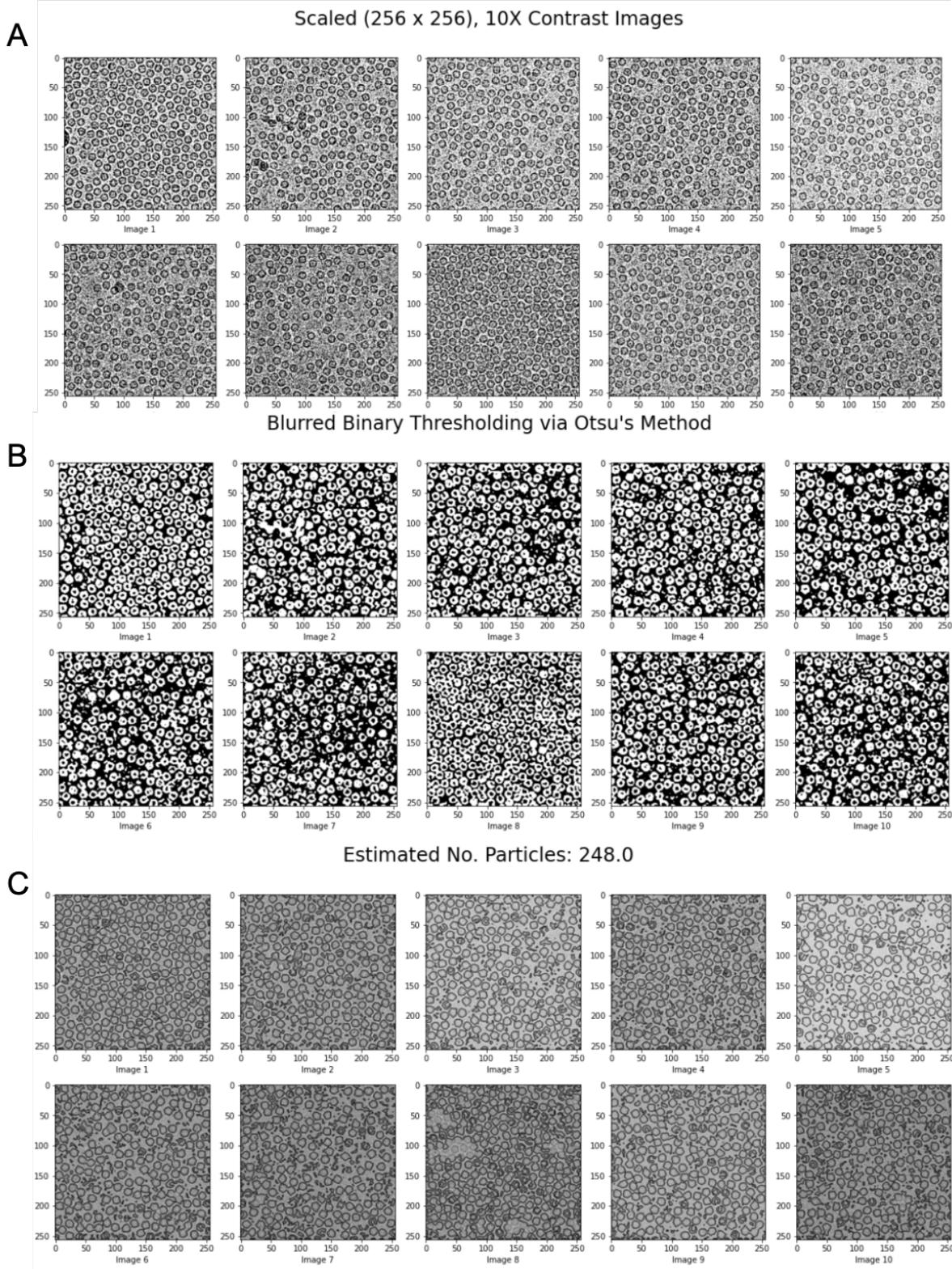


Figure 1: Top 10 Gaussian disk cross-correlated and defocused images from EMPIAR-10216 (mouse heavy-chain apoferritin at 1.62 angstrom). **(A)** Re-scaled images to 256 x 256, and 10X contrast increase. **(B)** Images are blurred with a 7x7 Gaussian kernel and Otsu's thresholding is applied to create a binary map. **(C)** Contours are extracted from the binary map. Only contours of hierarchy 1 are drawn to avoid contours inside apoferritin.

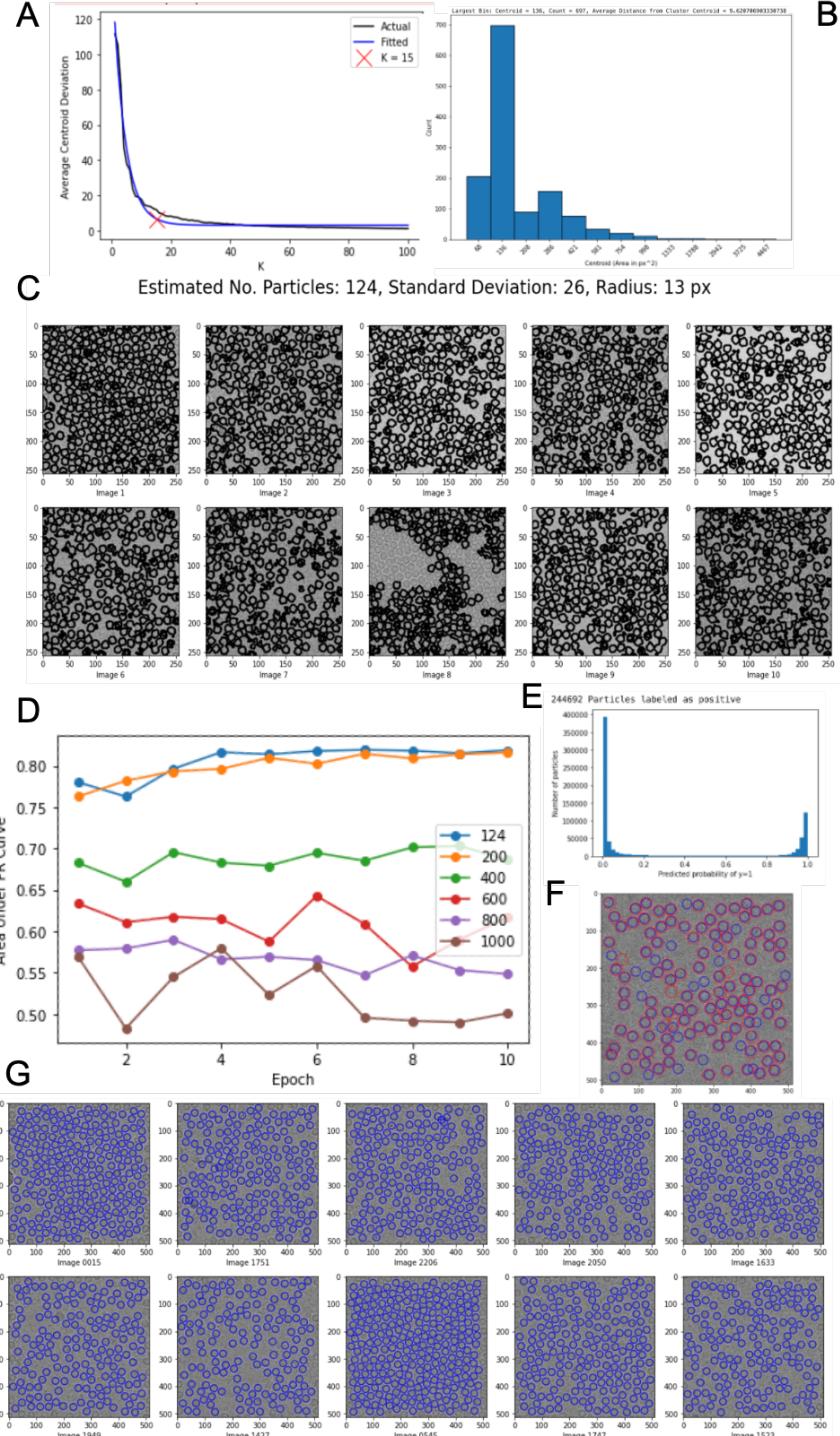


Figure 2: **(A)**  $K$ -means representation of contour area distributions. As  $K$  increases, the average centroid deviation decreases for 10216. We fit this curve with an exponential model, and a  $K = 15$  lead to  $\epsilon \leq 1$ , reaching our termination condition. We believe  $K = 15$  gives the best representation of the underlying particle stack. **(B)** Histogram distribution after first and last bins are removed. A peak in the bin with centroid of 136 indicates true particles. Tight cluster distribution of 9.62 indicates particles in bin are relatively homogeneous. **(C)** Predicted particle locations, density, and size after pruning with histogram. **(D)** Topaz test data set performance (AUPRC) after model is trained for 10 epochs with initial density estimation of 124,200,400,600,800,1000. **(E)** Using an initial estimate of 124, 244692 positive particles were found from 142218 labeled particles. **(F)** Particle locations extracted by Topaz. Red circles are ground truth (from EMPIAR), blue circles are Topaz selected. **(G)** Topaz particle picks on 10 images used for initial density and size estimation.

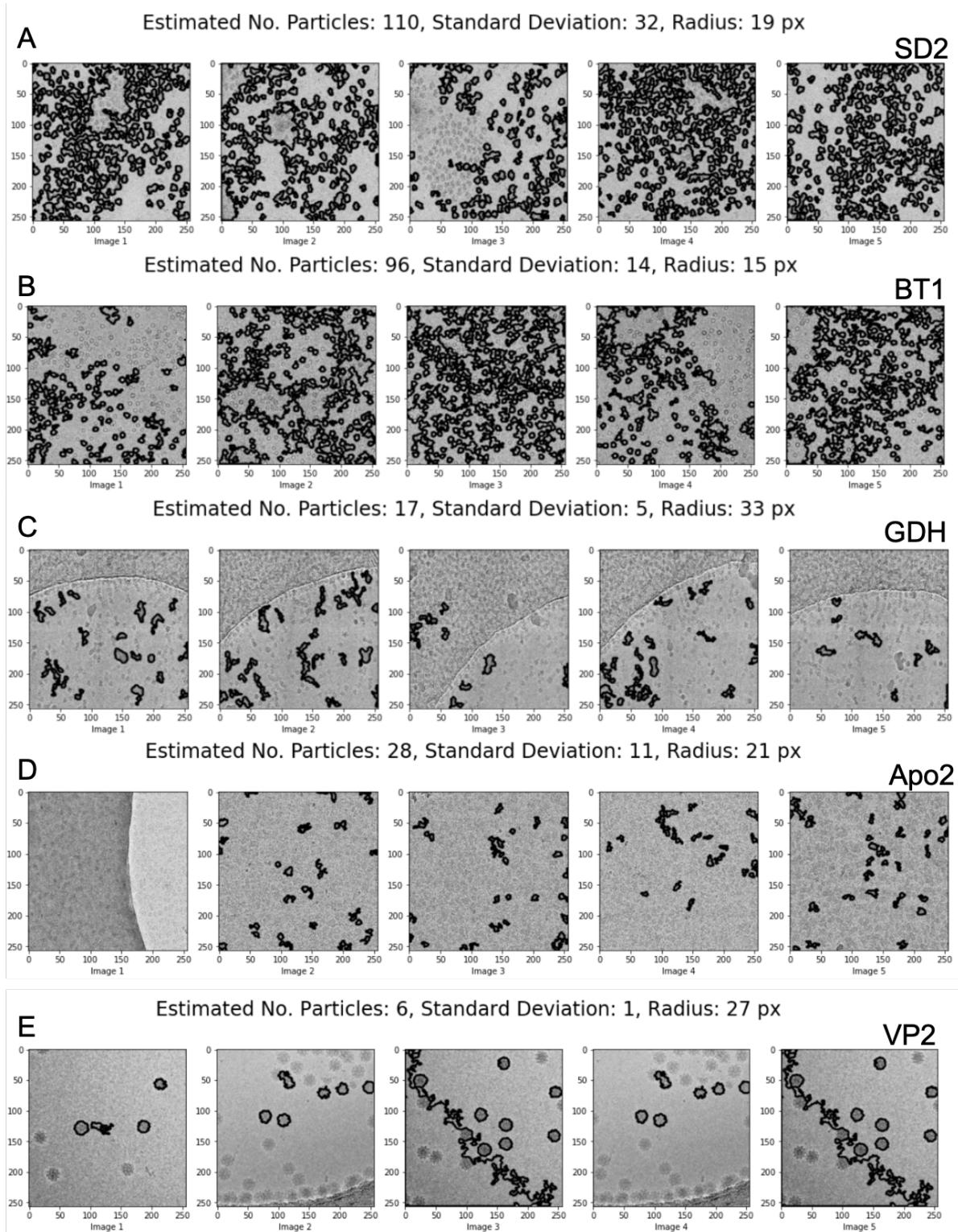


Figure 3: (A,B,C,D,E) Preprocessing pipeline estimations on Defensin SD2 (A), Adenine Nucleotide Transporter (B), Glutamate Dehydrogenase (C), Chloroplastic Precursor Apo2 (D), and EMPIAR-10192 Calcivirus VP2 (E).

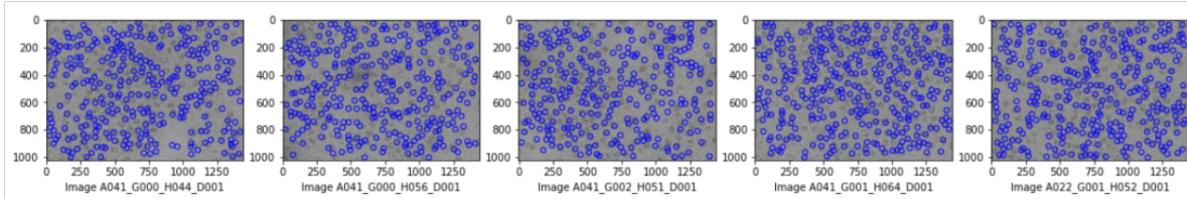
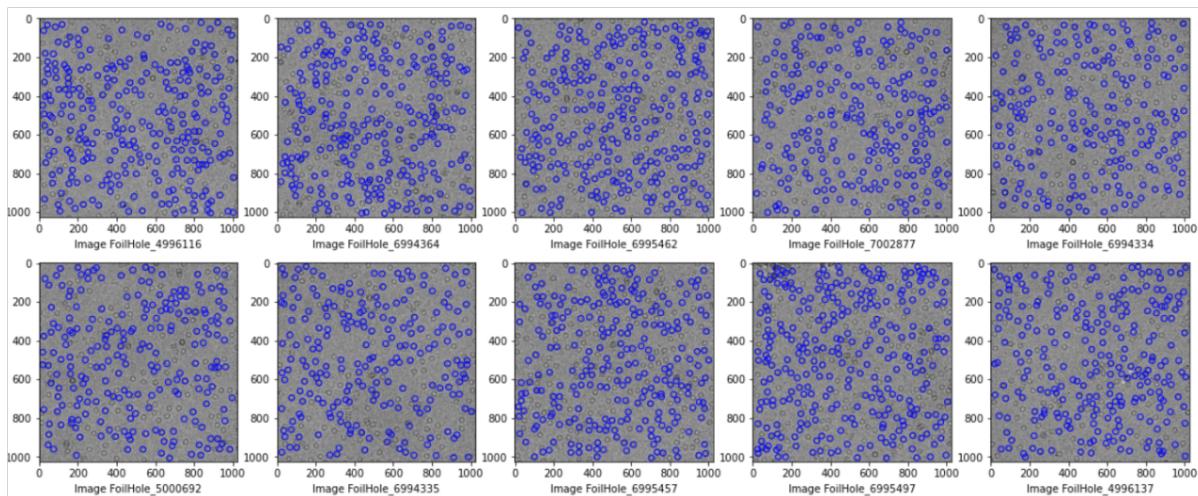
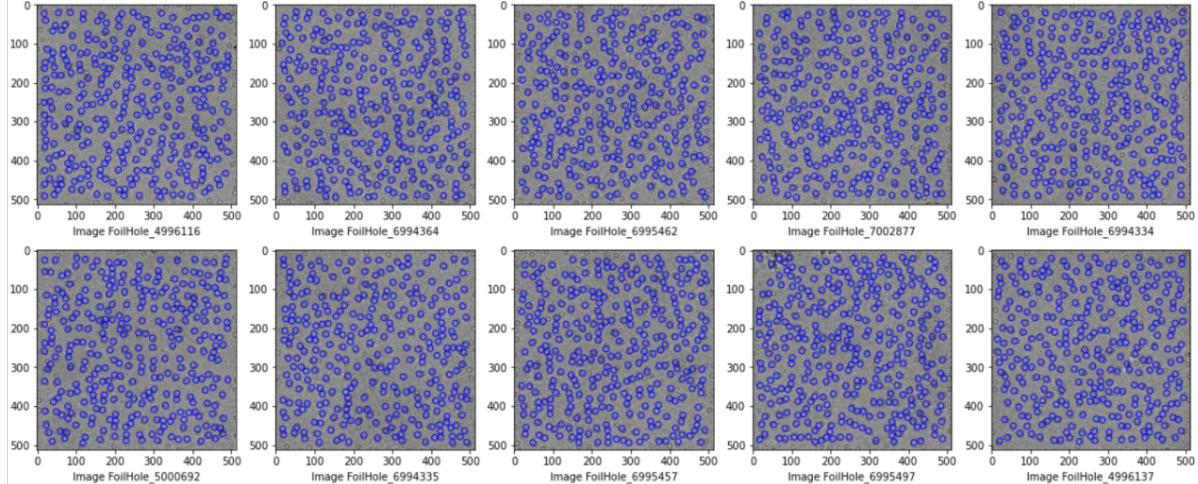
**A****B****C**

Figure 4: After training on EMPIAR-10216; (A) Topaz extracted particle positions on SD2 without initial image rescaling. (B) Topaz extracted particle positions on BT1 without initial image rescaling. (C) Topaz extracted particle positions on BT1 with initial image rescaling to 512 x 512.