Supplementary Information (SI) for Digital Discovery.

SUPPORTING INFORMATION

Machine Learning for Analyzing Atomic Force Microscopy (AFM) Images Generated from Polymer Blends

Aanish Paruchuri [1],Yunfei Wang[2], Xiaodan Gu[2], Arthi Jayaraman [3,4,5] *

1. Master of Science in Data Science Program, University of Delaware, Newark DE 19713
2. School of Polymer Science and Engineering, 118 College Drive, #5050 University of Southern Mississippi, Hattiesburg, MS 39406
3. Department of Chemical and Biomolecular Engineering, 150 Academy St, University of Delaware, Newark DE 19713
4. Department of Materials Science and Engineering, University of Delaware, Newark DE 19713
5. Data Science Institute, University of Delaware, Newark DE, 19713

*** Corresponding authors arthij@udel.edu**

S.I. Dataset

A. Link to raw data

Raw data used in this study has been deposited in Zenodo DOI: https://zenodo.org/doi/10.5281/zenodo.11179874

B. Visualization of noise in dataset prior and post denoising

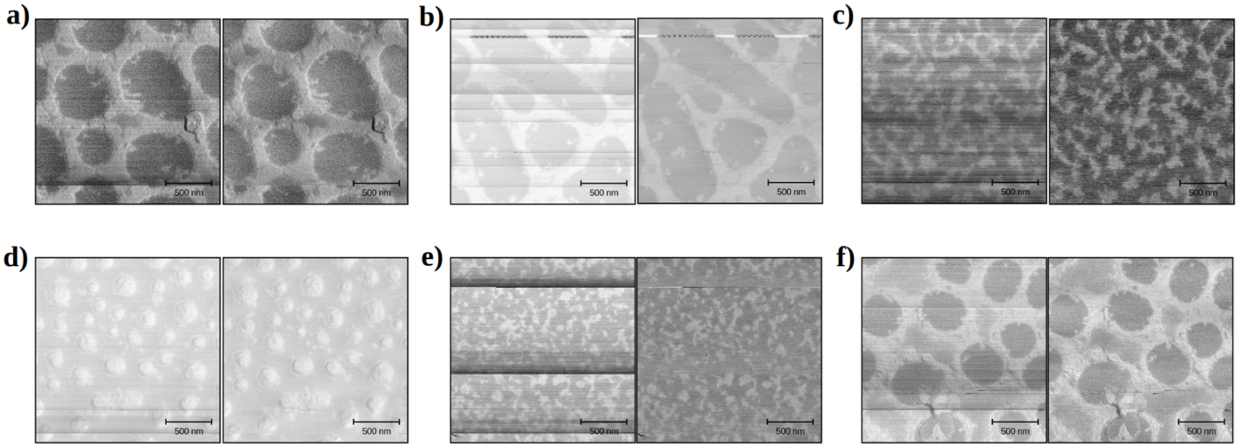


Figure S1: *Image pre-processing results. (a-f) For the presented six AFM images, the image on the left corresponds to the noisy AFM phase measurement and the image on the right corresponds to the denoised image. Image denoising operations are performed by Gwyddion software [1].*

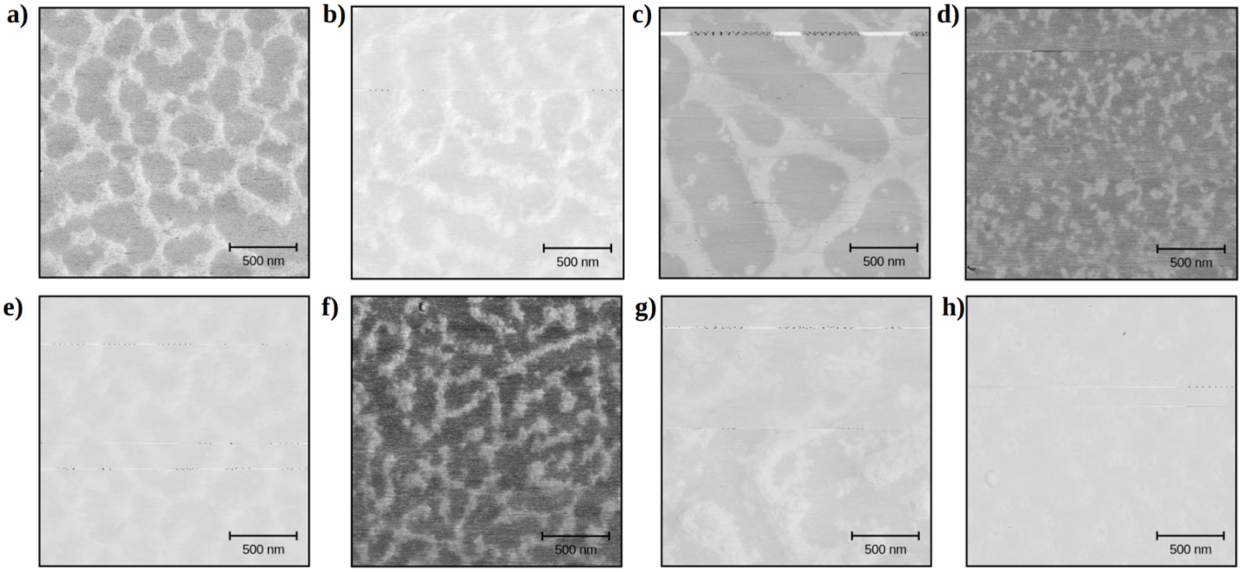


Figure S2: *(a-h) Pre-processed AFM phase images after denoising with Gwyddion[1] software; these images are shown to demonstrate that even after denoising these images retain some noise (visible defects).*

C. Thresholding techniques applied on dataset

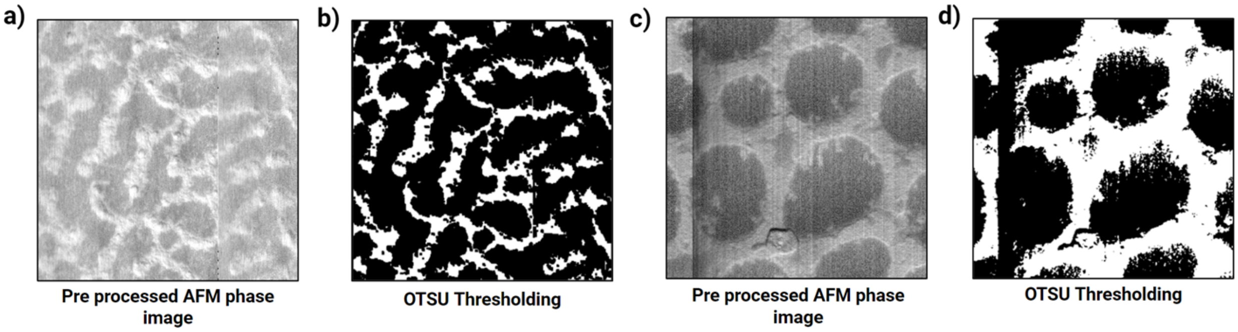


Figure S3: *OTSU thresholding algorithm applied on (a,c) pre-processed AFM phase images after denoising. (b,d) are their respective outputs when OTSU is applied locally on patches of the pre-processed AFM phase images and the denoised with image morphology operation. Thresholding fails in (d) due to the line noise and (b) lacks consistency in predicting continuity.*

S.II. Tiles and Win Factor

A. How to choose the best *win factor*

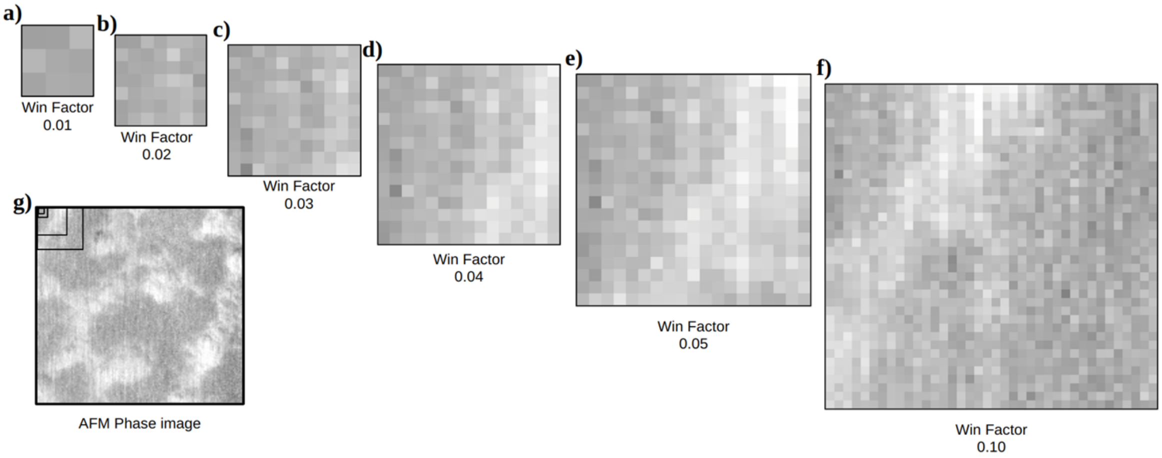


Figure S4: *(a-f) Samples of tiles with varying win factor from 0.01 to 0.10 generated from one pre-processed AFM phase image shown in part (g). To find the best tile size, we increase the win factor iteratively and choose the minimum size that can distinguish between light and dark domains in the AFM image. We can see that in parts (a) and (b) with low win factors 0.01 and 0.02 one cannot distinguish between the two domains. We notice that in part (c) the win factor of 0.03 presents a visual difference in light and dark domains. Therefore, we choose this as the best win factor. To confirm the details are of domains and not noise, we also generate for the reader the images in parts (d-f) with larger win factors.*

B. Pseudo code of generating *tiles* from input image with given *win factor*

A raw image is sub-sampled into tiles with the following workflow using steps below:

$$tile\_width = input\_img\_width * win\_factor \qquad (1)$$

$$tile\_height \; = \; input\_img\_height * win\_factor \qquad (2)$$

$$T\_ij = input\_img\_img[i \; - \frac{tile\_width}{2} \; :i + \; \frac{tile\_width}{2}, j - \frac{tile\_height}{2} :j \; + \frac{tile\_height}{2} \,]$$

$where,$

$$\mathrm{i} \; = \; (\frac{tile\_width}{2}),(\frac{\mathrm{tile\_width}}{2} + \mathrm{stride}), \; (\frac{\mathrm{tile\_width}}{2} \; + 2 * \mathrm{stride}), \; ... \; , \; (\mathrm{input\_img\_height} \; - \frac{\mathrm{tile\_width}}{2})$$

$$\mathrm{j} \; = \; (\frac{tile\_height}{2}),(\frac{\mathrm{tile\_height}}{2} + \mathrm{stride}), \; (\frac{\mathrm{tile\_height}}{2} \; + 2 * \mathrm{stride}), \; ... \; , \; (\mathrm{input\_img\_height} \; -$$

(3)

Variables *stride* and *win factor* have a direct impact on feature extraction and are also responsible for the resolution of domain segmentation output. *Stride* has values starting at 1. As we increase *stride*, it decreases the resolution in prediction of domain segmentation output. The *win factor* is responsible for the tile size and a large tile size results in more boundary pixels excluded from analysis (the latter is described in main manuscript's **section II A).**

S.III. Discrete Wavelet Transform (DWT)

A. Performance and characteristics of different wavelet types in DWT workflow

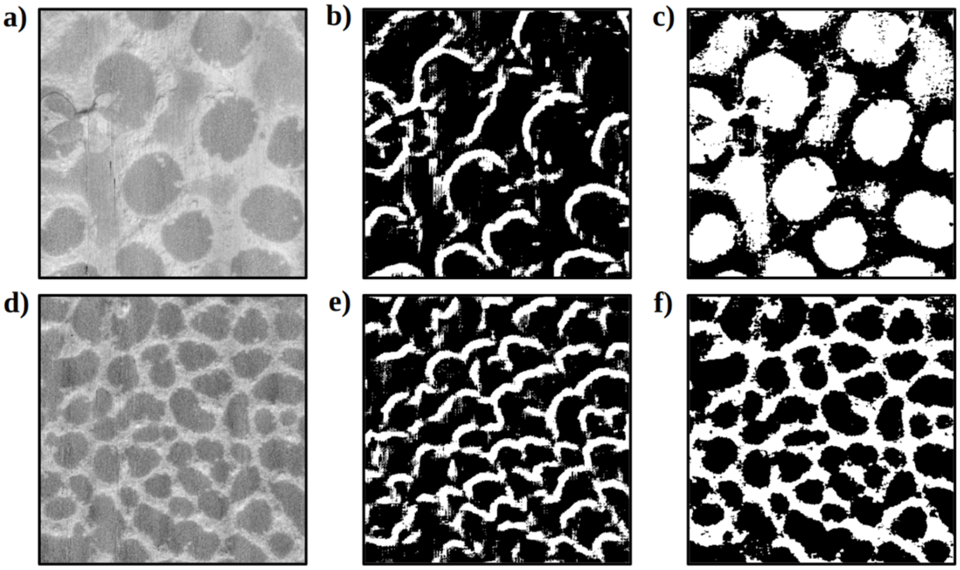


Figure S5: Parts *(a) and (d) are two preprocessed AFM phase images and parts (b, c) and parts (e, f) are their domain segmentation outputs, respectively. The DWT workflow's domain segmentation outputs from (b, e) Haar wavelet and (c, f) biorthogonal wavelets are shown. Depending on the convolutional filters inherited by the type of wavelet, each has a different characteristic of decomposition. (b, e) Haar wavelets focus more on capturing larger gradients whereas (c, f) biorthogonal wavelets capture more continuous gradients. As a result, we get to see different domain segmentations in both the cases (b, e) and (c, f).*

B. Extension of DWT workflow on other literature datasets: scope and opportunities

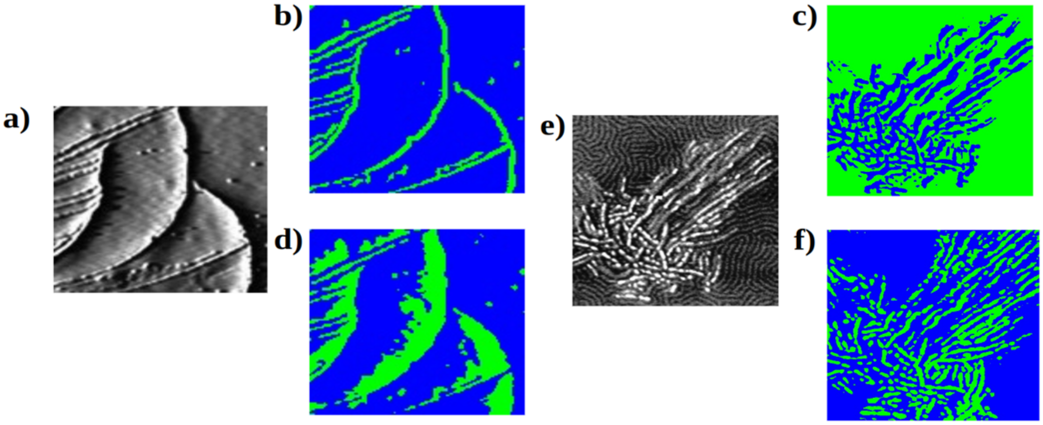


Figure S6: *DWT methods applied to other AFM images (parts a and e) adapted with permission from [2, 3] Copyright 2001 American Chemical Society. The images in (b, c) and (d, f) are the corresponding Haar*

and bioorthogonal domain segmentations for parts a and e, respectively. The domain segmentations are useful to study fibril like patterns in AFM images where features like length, directionality, and orientation of fibrils are of interest.

S.IV. Radon Transform

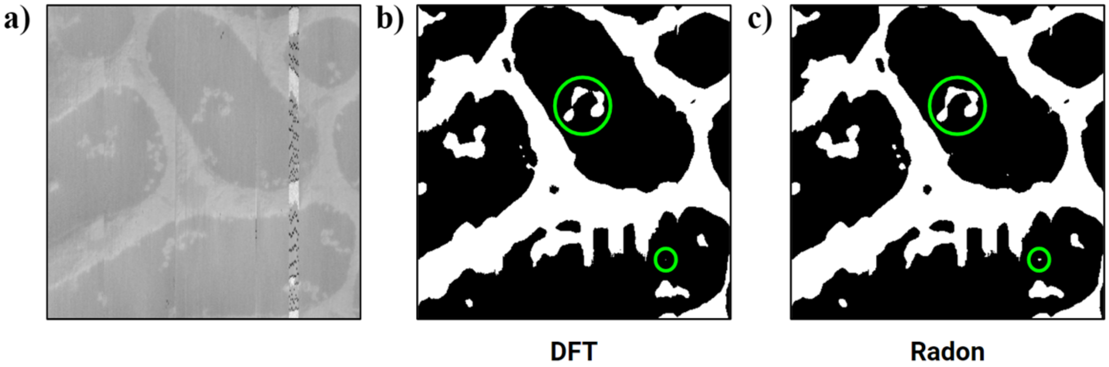


Figure S7: *DFT and Radon workflow applied on (a) a pre-processed AFM phase image resulting in (b, c) domain segmentations. In parts (b) and (c) the green circles highlight the minor differences in segmentation obtained from these two workflows.*

S.V. ResNet50: Methods experimented to improve ResNet50 performance

1. Methods to improve performance on noisy data.

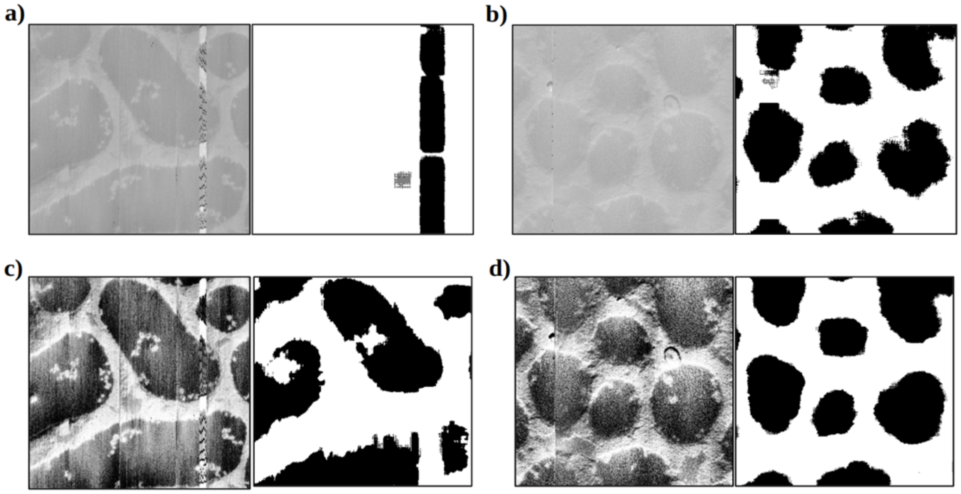


Figure S8: *ResNet50 workflow applied on pre-processed AFM phase images shown on the left in parts (a) and (b) without histogram equalization results in the right image in parts (a) and (b). As we can see the results are prone to noise in the pre-processed AFM image as ResNet50 is sensitive to scale and outliers in the image. Applying histogram equalization to the preprocessed AFM phase images before using the ResNet50 workflow shows tremendous improvement in the results as shown in parts (c) and (d).*

2. Methods to address minimum tile size.

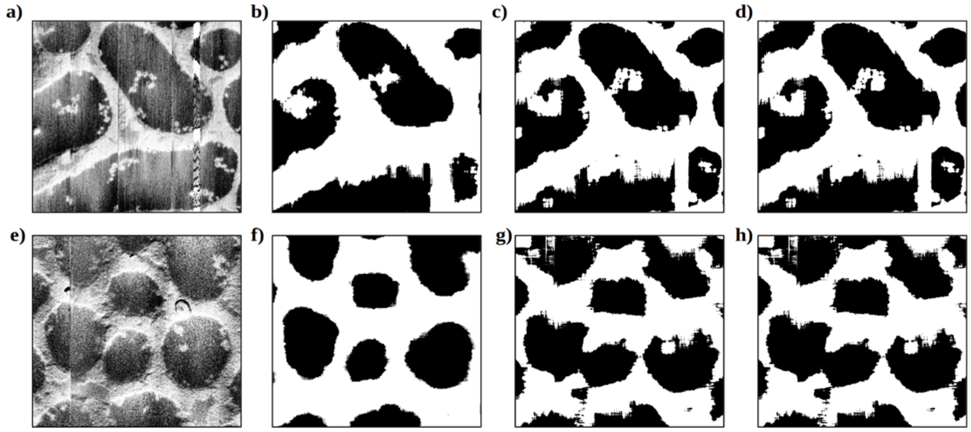


Figure S9: *To address the input tile size constraint discussed in the main manuscript's* ***Section II. B****, one could increase the overall image size with image interpolation techniques which can increase the scales of features in tiles. In parts (a) and (e) we show two histogram equalized pre-processed AFM phase images that are sent into the ResNet50 workflow yielding the domain segmented images in parts (b-d) and parts (f-h) when the pre-processed images were interpolated to sizes (c, g) 800 pixels x 800 pixels, (b, f) kept same as input 384 pixels x 384 pixels, and (d, h) 1200 pixels x 1200 pixels. We notice that interpolation has increased the workflow's ability to capture highly granular features (light domains inside larger dark domains are captured in parts c and d as compared to part b. We note, however, that the interpolation method's computational cost scales exponentially with increase in interpolation size.*

3. Understanding the use of ResNet 50 architecture to extract features

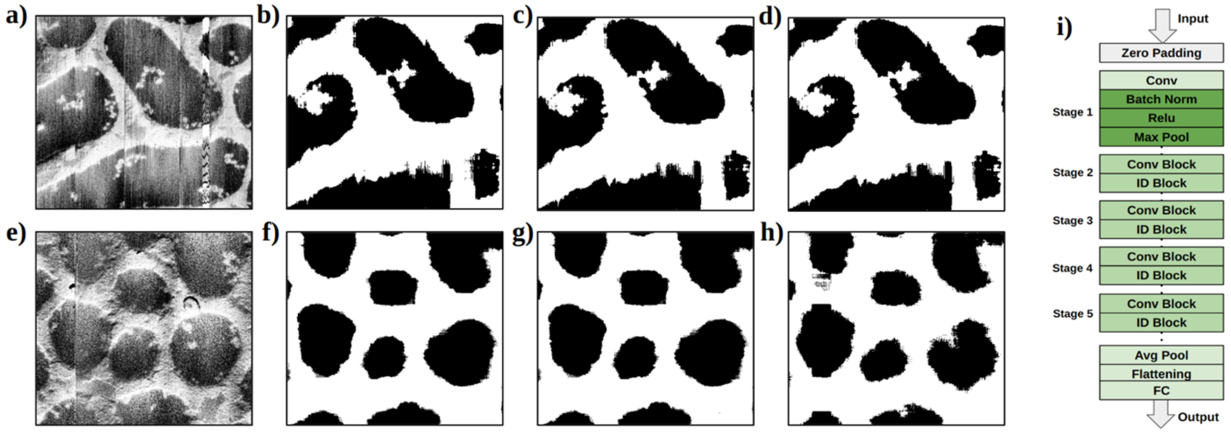


Figure S10: *ResNet50 is a deep learning algorithm with multiple layers. For the input histogram equalized preprocessed AFM phase images shown in parts (a) and (e), we show the domain segmentation images generated by using feature maps from various stages of ResNet50 model shown in part (i). In parts (b) and (f) are the results for the two inputs when we used feature maps from stage 1 of the flow in part (i). In parts (c) and (g) are results using stage 2 feature maps and in parts (d) and (h) are results using stage3 features maps. With increase in depth, we notice that the workflow becomes more prone to noise [e.g., you can see noise in part (h) that we do not see in parts (f) and (g)]; there can also be misclassification in boundary regions of domains.*

S.VI. Domain Size Distribution Calculated Using Porespy Package

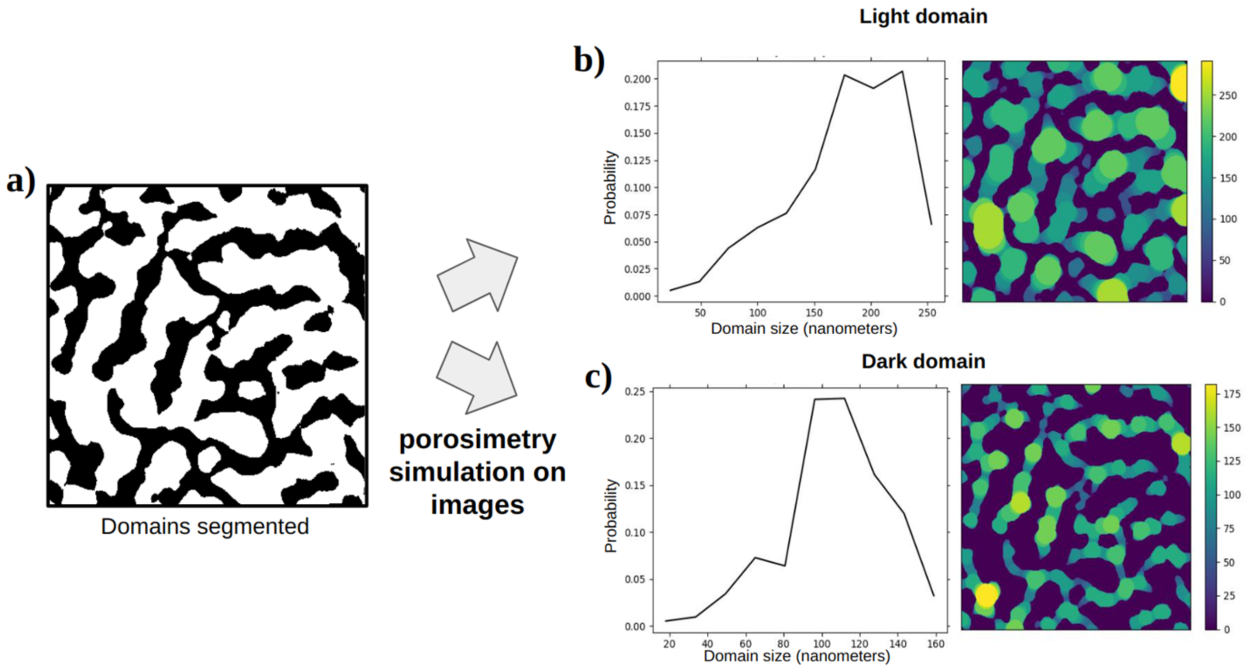
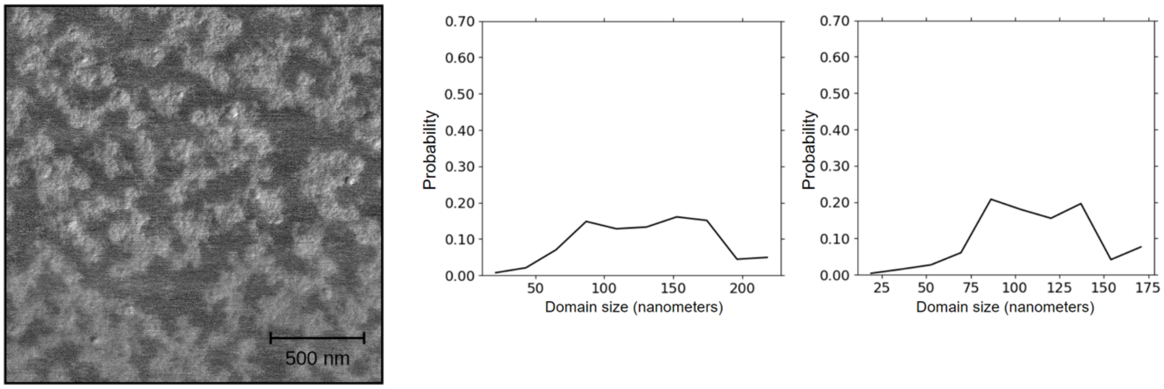


Figure S11: *Illustration of domain size distribution calculations. (a) Index maps (binary images) containing domain segmented into light and dark domains are used to perform domain size distribution calculation.* ***Porespy [4] is a python package*** *that performs porosimetry simulations on index maps resulting in 2D heat maps as shown in the right side of parts (b) and (c). In these heat maps the color of each pixel depicts the radius of the largest circle that could overlap that pixel and the pixels of the non-observing domain are zero. From the heat maps pixel values, we can then calculate domain size distribution in real units using the scale bar present in the metadata associated with the AFM image used in the workflow; parts (b) and (c) on the left present these distributions.*

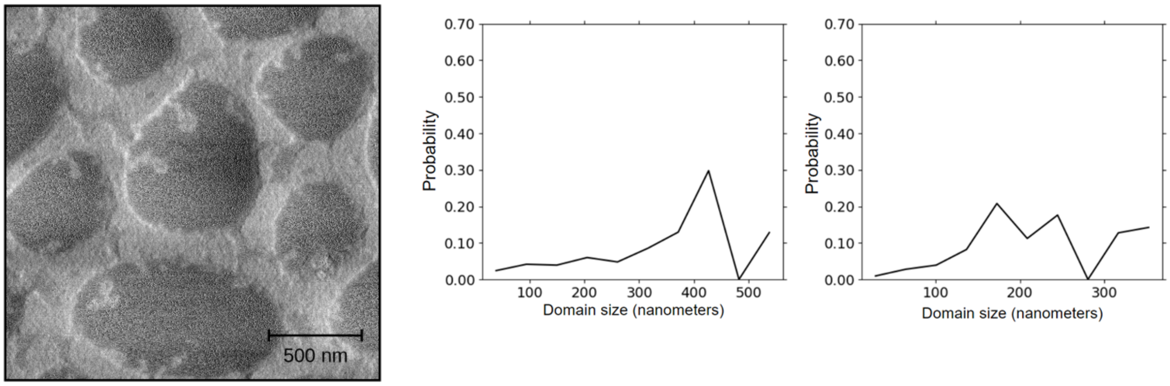
In **Figure S12** we present the calculated domain size distributions for representative 15 AFM images from the 144 images we had in the dataset. All of these 15 AFM images were segmented using DFT with variance as the features. As described before, these AFM images were obtained from supramolecular block copolymer with varying PS and POEGMA block lengths.

Pre-processed image	**Light domain**	**Dark domain**

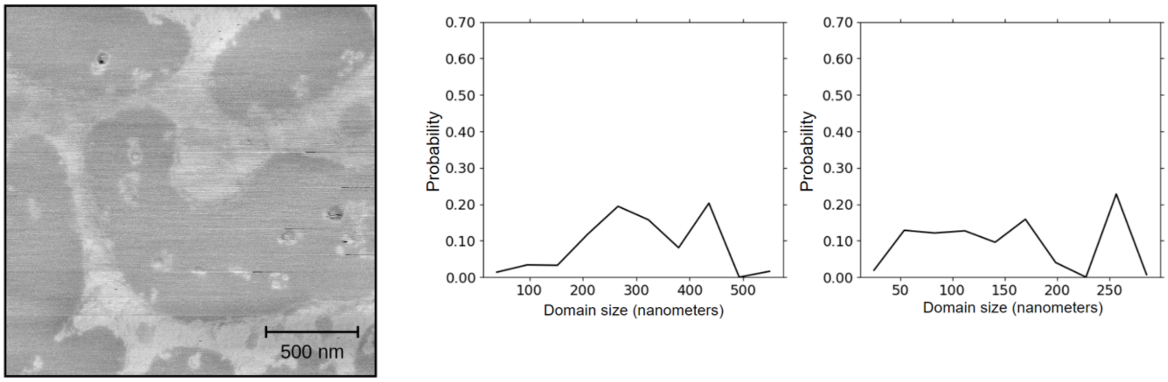
a).



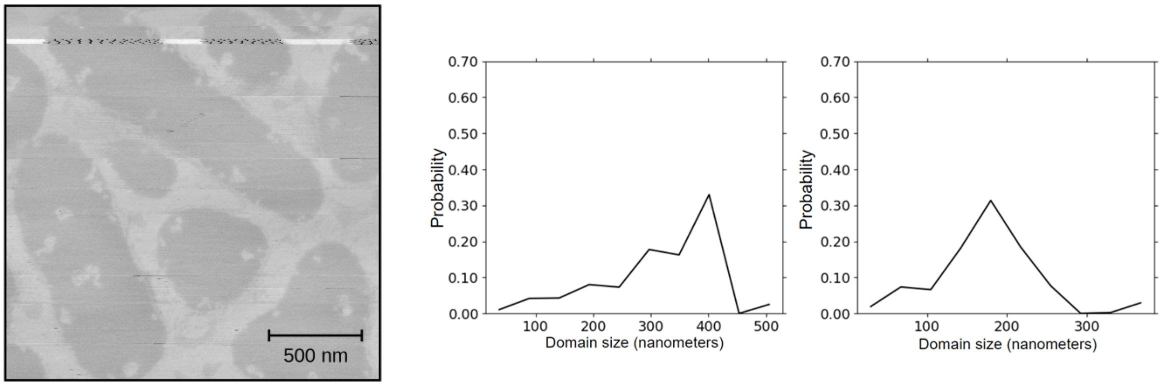
b).



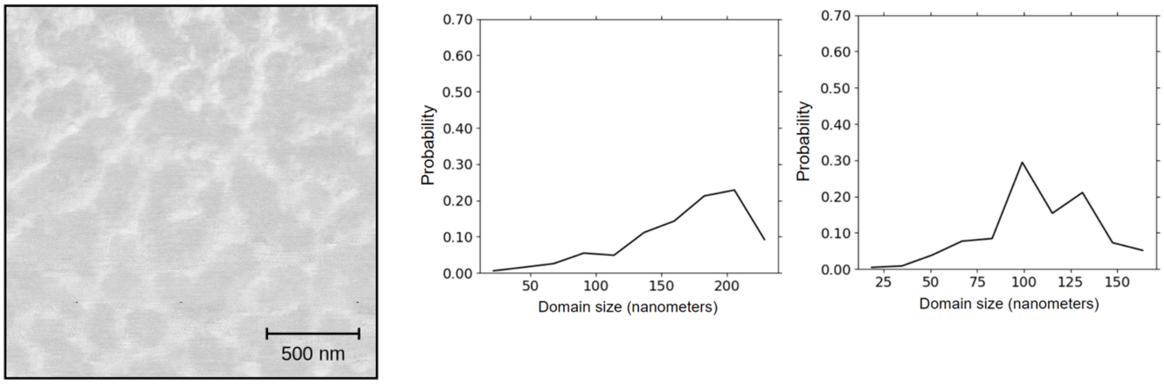
c).



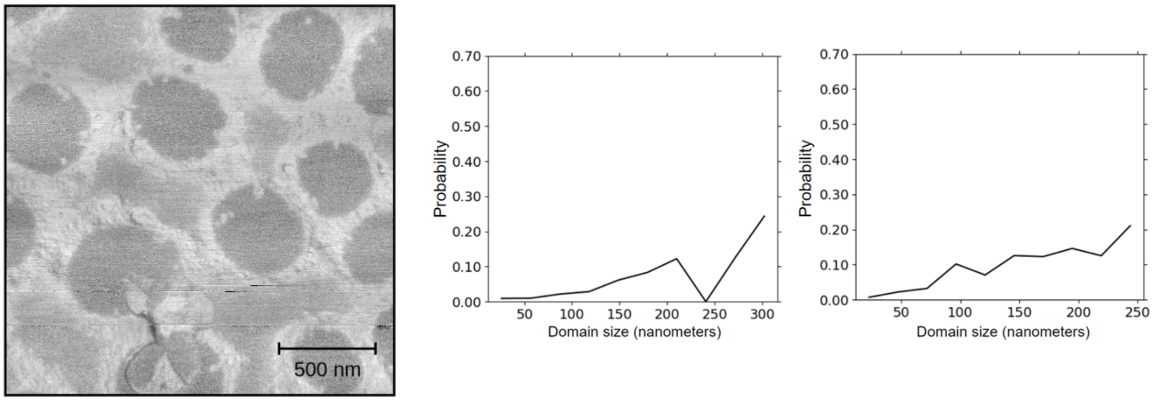
d).


500 nm
Probability
0.00
0.10
0.20
0.30
0.40
0.50
0.60
0.70
100
200
300
400
500
Domain size (nanometers)
Probability
0.00
0.10
0.20
0.30
0.40
0.50
0.60
0.70
100
200
300
Domain size (nanometers)

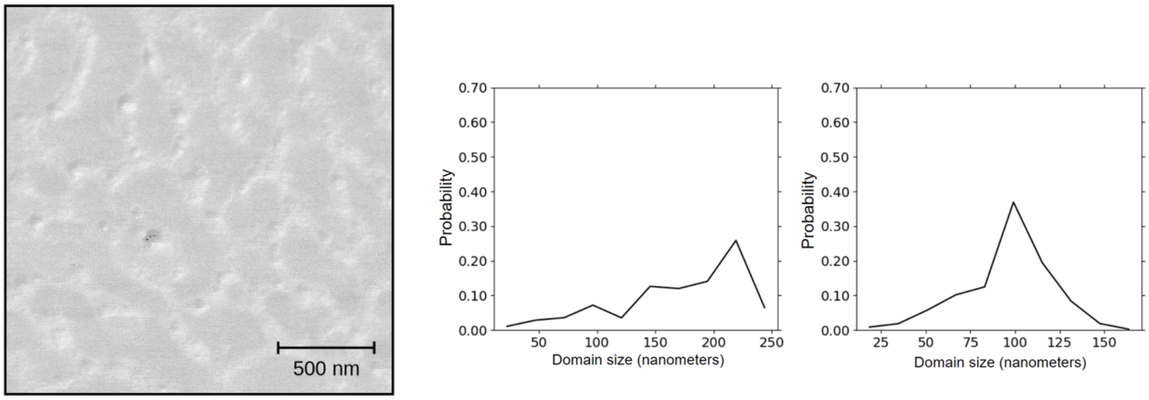
e).


500 nm
Probability
0.00
0.10
0.20
0.30
0.40
0.50
0.60
0.70
50
100
150
200
Domain size (nanometers)
Probability
0.00
0.10
0.20
0.30
0.40
0.50
0.60
0.70
25
50
75
100
125
150
Domain size (nanometers)

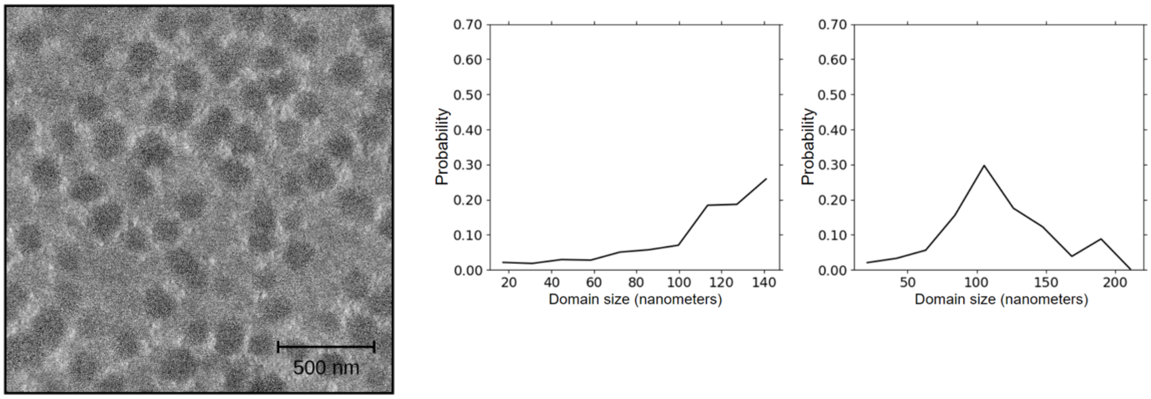
f).


500 nm
Probability
0.00
0.10
0.20
0.30
0.40
0.50
0.60
0.70
50
100
150
200
250
300
Domain size (nanometers)
Probability
0.00
0.10
0.20
0.30
0.40
0.50
0.60
0.70
50
100
150
200
250
Domain size (nanometers)

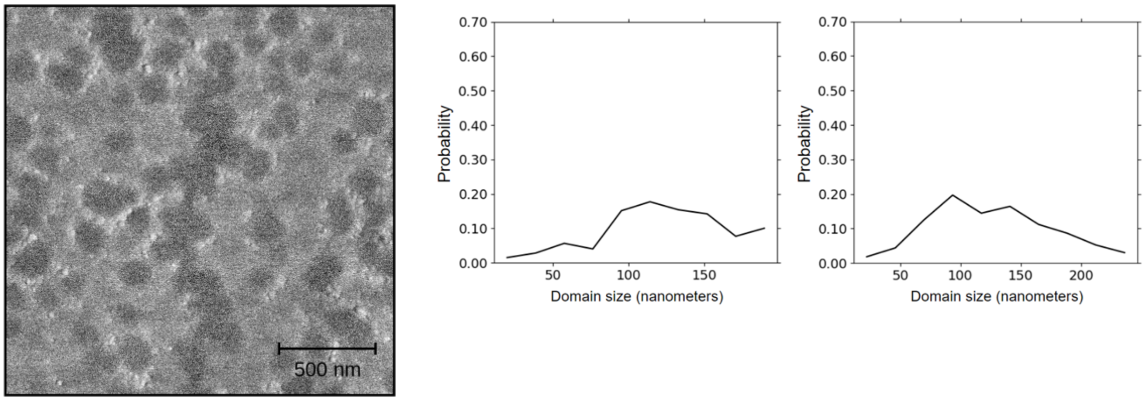
g).


500 nm
Probability
0.70 0.60 0.50 0.40 0.30 0.20 0.10 0.00
50 100 150 200 250
Domain size (nanometers)
Probability
0.70 0.60 0.50 0.40 0.30 0.20 0.10 0.00
25 50 75 100 125 150
Domain size (nanometers)

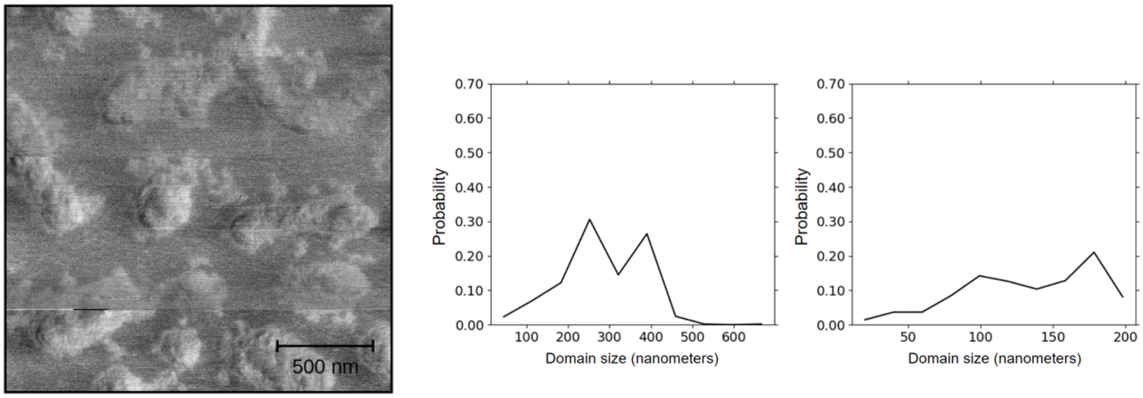
h).


500 nm
Probability
0.70 0.60 0.50 0.40 0.30 0.20 0.10 0.00
20 40 60 80 100 120 140
Domain size (nanometers)
Probability
0.70 0.60 0.50 0.40 0.30 0.20 0.10 0.00
50 100 150 200
Domain size (nanometers)

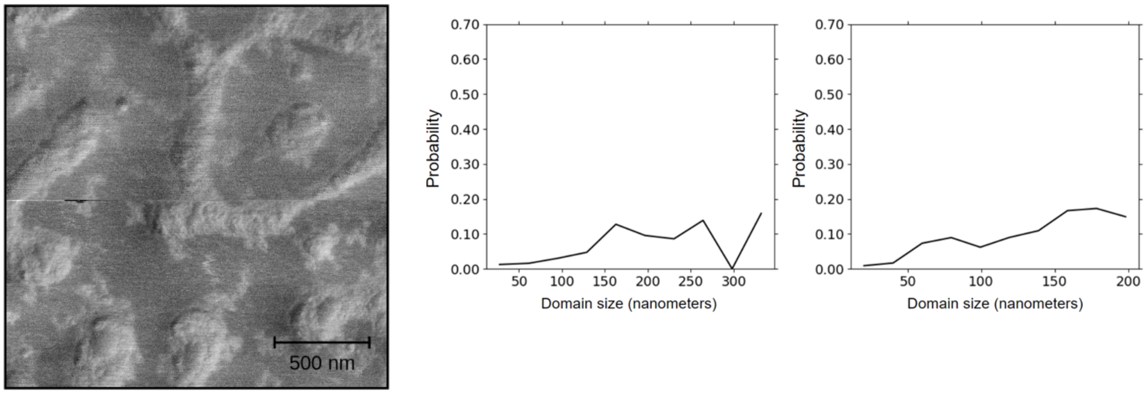
i).


500 nm
Probability
0.70 0.60 0.50 0.40 0.30 0.20 0.10 0.00
50 100 150
Domain size (nanometers)
Probability
0.70 0.60 0.50 0.40 0.30 0.20 0.10 0.00
50 100 150 200
Domain size (nanometers)

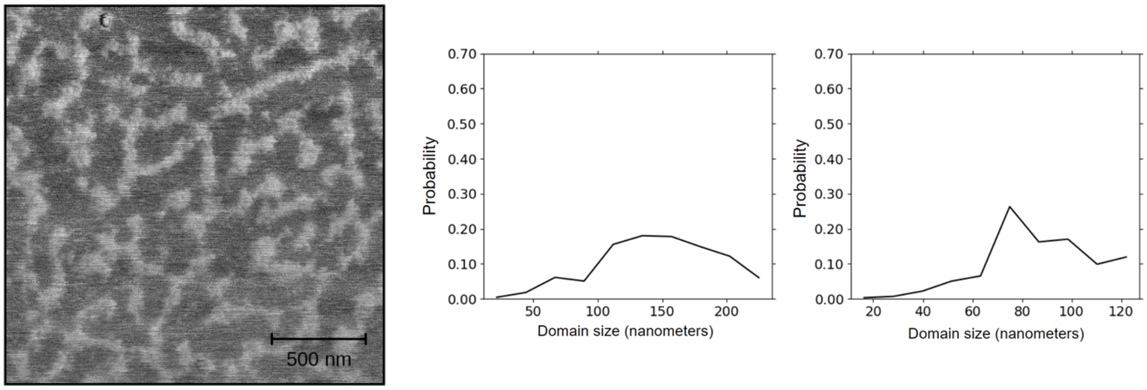
j).


500 nm
Probability
0.00
0.10
0.20
0.30
0.40
0.50
0.60
0.70
100
200
300
400
500
600
Domain size (nanometers)
Probability
50
100
150
200
Domain size (nanometers)

k).


500 nm
Probability
0.00
0.10
0.20
0.30
0.40
0.50
0.60
0.70
50
100
150
200
250
300
Domain size (nanometers)
Probability
50
100
150
200
Domain size (nanometers)

l).


500 nm
Probability
0.00
0.10
0.20
0.30
0.40
0.50
0.60
0.70
50
100
150
200
Domain size (nanometers)
Probability
20
40
60
80
100
120
Domain size (nanometers)

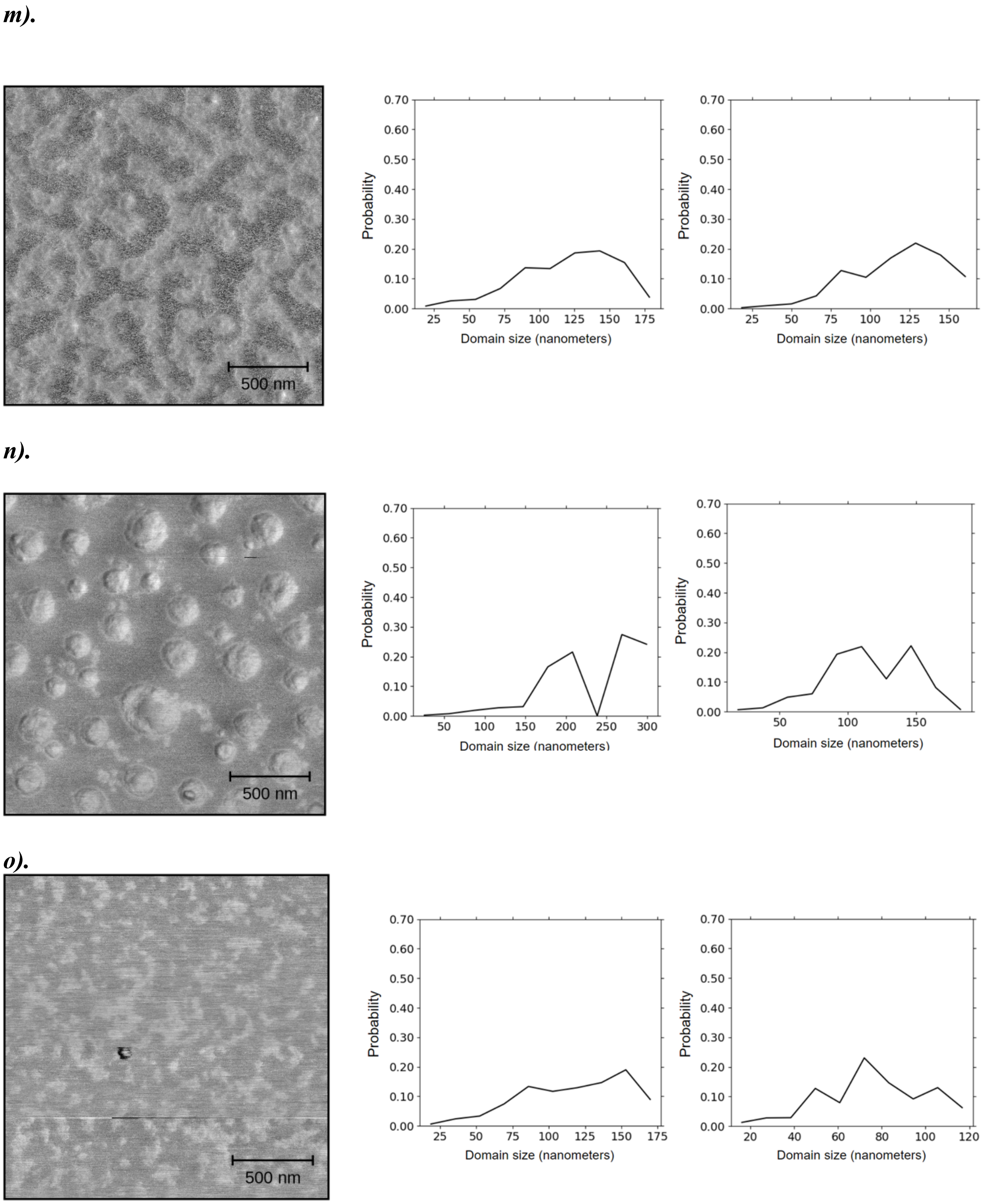
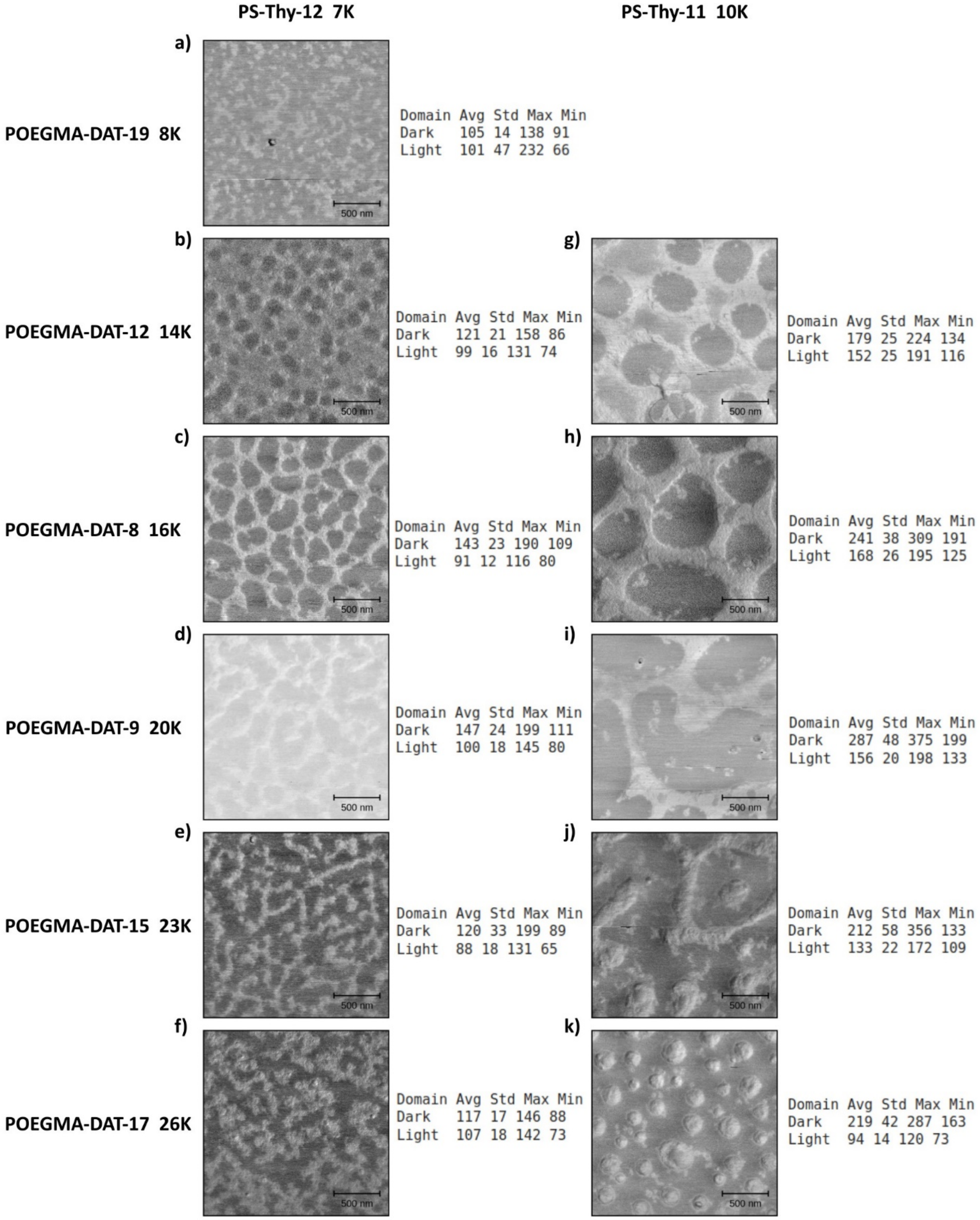


***Figure S12*. *Results of domain size distribution for another 15 representative AFM images*.** *Each panel we have two figures – left is the original AFM pre-processed image and on the right is the domain size distributions (Probability vs. domain sizes in nm) of dark and light domains.*



a)

```
Domain Avg Std Max Min
Dark   105 14 138 91
Light  101 47 232 66
```

b)

```
Domain Avg Std Max Min
Dark   121 21 158 86
Light  99 16 131 74
```

c)

```
Domain Avg Std Max Min
Dark   143 23 190 109
Light  91 12 116 80
```

d)

```
Domain Avg Std Max Min
Dark   147 24 199 111
Light  100 18 145 80
```

e)

```
Domain Avg Std Max Min
Dark   120 33 199 89
Light  88 18 131 65
```

f)

```
Domain Avg Std Max Min
Dark   117 17 146 88
Light  107 18 142 73
```

g)

```
Domain Avg Std Max Min
Dark   179 25 224 134
Light  152 25 191 116
```

h)

```
Domain Avg Std Max Min
Dark   241 38 309 191
Light  168 26 195 125
```

i)

```
Domain Avg Std Max Min
Dark   287 48 375 199
Light  156 20 198 133
```

j)

```
Domain Avg Std Max Min
Dark   212 58 356 133
Light  133 22 172 109
```

k)

```
Domain Avg Std Max Min
Dark   219 42 287 163
Light  94 14 120 73
```

Figure S13. *Summary of results from domain size distributions for various supramolecular polymers in the AFM dataset shown in* ***Figure S12***. *Here we show the value of the molecular weights of the POEGMA-DAT and PS-Thy in the leftmost column and top-most row. In each part* *on the left we show the representative AFM image for that system. On the right we share the average, standard deviation, maximum, and minimum of the domain sizes seen for various images collected for each sample with the corresponding molecular weights of the POEGMA-DAT and PS-Thy.*

References

1. Nečas, D. and P. Klapetek, *Gwyddion: an open-source software for SPM data analysis.* Open Physics, 2012. **10**(1): p. 181-188.
2. Hobbs, J.K., A.D.L. Humphris, and M.J. Miles, *In-Situ Atomic Force Microscopy of Polyethylene Crystallization. 1. Crystallization from an Oriented Backbone.* Macromolecules, 2001. **34**(16): p. 5508-5519.
3. Hobbs, J.K. and R.A. Register, *Imaging Block Copolymer Crystallization in Real Time with the Atomic Force Microscope.* Macromolecules, 2006. **39**(2): p. 703-710.
4. Gostick, J.T., et al., *PoreSpy: A python toolkit for quantitative analysis of porous media images.* Journal of Open Source Software, 2019. **4**(37): p. 1296.