

**Supporting information for:**

**Automated Quantitative Image Analysis of**

**Nanoparticle Assembly**

Chaitanya R. Murthy, Bo Gao, Andrea Tao, and Gaurav Arya\*

*Department of NanoEngineering*  
*University of California, San Diego*  
*9500 Gilman Drive, Mail Code 0448*  
*La Jolla, CA 92093*

E-mail: [garya@ucsd.edu](mailto:garya@ucsd.edu)

Phone: 858-822-5542. Fax: 858-534-9553

---

\*To whom correspondence should be addressed

## Unbiased feature measurement

This section provides a derivation of the exact formula used to calculate bias-correction weights in our software. We begin with Eq. 1 from the main text, which gives the conditional probability that an object of fixed orientation will entirely fit within a randomly placed field of view (and will therefore be measurable), given that its centroid is contained in that field of view:

$$P = \frac{(W_x - F_x)(W_y - F_y)}{W_x W_y} \quad (\text{S1})$$

where  $W_x$  and  $W_y$  are the dimensions of the image in the  $x$  and  $y$  directions, and  $F_x$  and  $F_y$  are the maximum dimensions of the object in those directions. The extent of an object along some arbitrary direction  $\theta$  is given by

$$F_\theta = \max_k (\vec{r}_k \cdot \hat{e}_\theta) - \min_k (\vec{r}_k \cdot \hat{e}_\theta) \quad (\text{S2})$$

where  $\vec{r}_k$  is the position vector of point  $k$  relative to the centroid of the object and  $\hat{e}_\theta$  is the unit vector specifying the direction; we define  $\theta$  as the counterclockwise angle from the positive  $x$ -axis, so that  $\hat{e}_\theta = [\cos \theta, \sin \theta]$ . The orientation-averaged conditional probability that an object  $i$  will be measurably contained within a randomly placed field of view is then given by

$$\langle P_{i\theta} \rangle_\theta = \frac{\langle (W_x - F_\theta)(W_y - F_{\theta+\frac{\pi}{2}}) \rangle_{\theta \in [0, \pi)}}{W_x W_y} \quad (\text{S3})$$

where the angle brackets denote averaging of the specified variable over the given range, and only angles smaller than  $\pi$  radians need to be considered in the average because  $F_{\theta+\pi} = F_\theta$ . Taking the inverse of both sides of Eq. S3 yields Eq. 2 in the main text. Eq. S3 can be rewritten as

$$\begin{aligned} \langle P_{i\theta} \rangle_\theta &= \frac{\langle (W_x - F_\theta)(W_y - F_{\theta+\frac{\pi}{2}}) + (W_x - F_{\theta+\frac{\pi}{2}})(W_y - F_\theta) \rangle_{\theta \in [0, \frac{\pi}{2}]} }{2W_x W_y} \\ &= \frac{\langle 2W_x W_y + 2F_\theta F_{\theta+\frac{\pi}{2}} - (W_x + W_y)(F_\theta + F_{\theta+\frac{\pi}{2}}) \rangle}{2W_x W_y} \\ &= \frac{W_x W_y + \langle F_\theta F_{\theta+\frac{\pi}{2}} \rangle - \frac{1}{2}(W_x + W_y)\langle F_\theta + F_{\theta+\frac{\pi}{2}} \rangle}{W_x W_y} \end{aligned} \quad (\text{S4})$$

where we have grouped terms and expanded the numerator, using the periodicity of  $F_\theta$  to again reduce the range of angles considered (all averages here are taken over  $0 \leq \theta < \pi/2$ ). The code assigns a weight of  $w_i = 1/\langle P_{i\theta} \rangle_\theta$  to each measured cluster, where  $\langle P_{i\theta} \rangle_\theta$  is calculated using Eq. S4.

## Empirical distribution of single-particle areas

The formula implemented in our software (Eq. 9 in main text) to estimate cluster size distributions via Bayesian inference makes use of the assumption that the distribution of single-particle areas is Gaussian. Figure S1 contains a histogram of all particle areas measured in our aggregation experiment involving Ag nanocubes (E1). The distribution of particle areas does appear to be very nearly Gaussian; similar results were obtained in the other experiments that we analyzed.

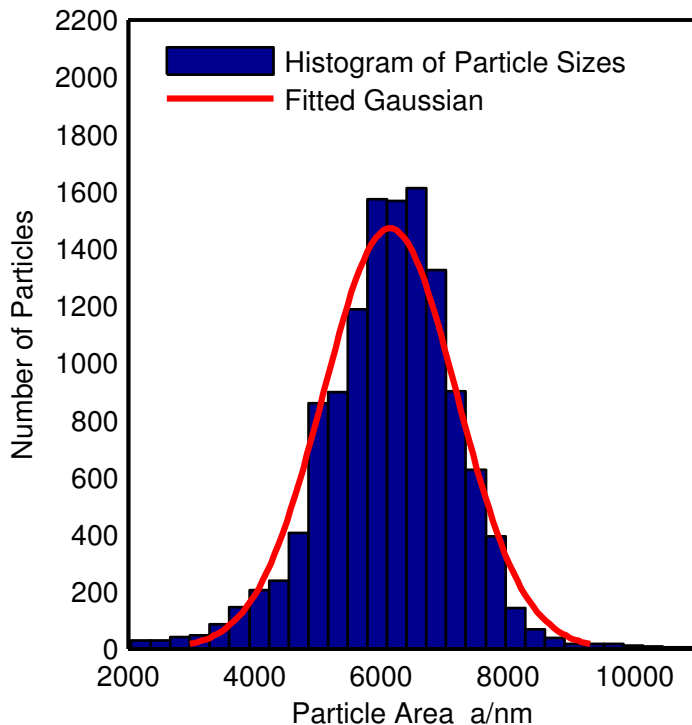


Figure S1: Histogram of particle areas for all particles identified in images taken during our Ag nanocube aggregation experiment (E1). The red curve is a fitted Gaussian.

## Calculation of self-similarity dimensions by regression on cluster data

As discussed in the main text, our software computes various self-similarity dimensions  $D$  by linear regression (LR) on  $\log(s)$  versus  $\log(l)$  data, where  $s$  is cluster size and  $l$  is some cluster length scale. The corresponding dimension is obtained directly as the slope of such a fit. An alternative would be to perform non-linear regression (NLR) on the original (not log-transformed) data, and obtain  $D$  or its inverse as the exponent in the fitted power law. Xiao et al. have compared these two methods (LR on log-transformed data versus NLR on the original data) and have demonstrated that the error distribution may be used to determine which method to use.<sup>S1</sup>

We have analyzed the distribution of residuals obtained in both kinds of regression on our cluster data. Figure S2 shows regressions aimed at computing the fractal dimension  $D_f^{(R)}$ , which describes the scaling of cluster size  $s$  with radius of gyration  $R_g$ :

$$s \sim R_g^{D_f^{(R)}} \quad (\text{S5})$$

The same original dataset is used in each regression, namely data for all clusters observed at time points  $t = 173$  and 180 minutes in our aggregation experiment involving Ag nanocubes (E1). In the regressions where  $R_g$  or  $\log(R_g)$  is on the  $x$ -axis, only data for clusters with  $R_g > 3r_1$  are plotted and used in the fits, where  $r_1$  is the mean single-particle radius of gyration for the same experiment. In the regressions where  $s$  or  $\log(s)$  is on the  $x$ -axis, only data for clusters of estimated size  $s \geq 4$  are plotted and used in the fits. The residuals from the LRs appear to be homoscedastic (Figure S2b,h) and approximately normally distributed (Figure S2c,i). By contrast, the residuals from the NLRs are clearly heteroscedastic (Figure S2e,k) and their distribution is not normal (Figure S2f,l). Similar results were obtained for other experiments, time points, and self-similarity dimensions.

These results strongly suggest that log-transformation and LR is preferable to NLR for computing self-similarity dimensions from cluster data. We further prefer fitting  $\log(s)$  versus  $\log(R_g)$  (Figure S2 a-c) rather than the opposite (Figure S2g-i) because in the former the slope of the fit directly gives the dimension. Similarly, other self-similarity dimensions are computed as the slopes of linear fits to  $\log(s)$  versus  $\log(l)$  data, where  $l$  is the appropriate cluster length scale.

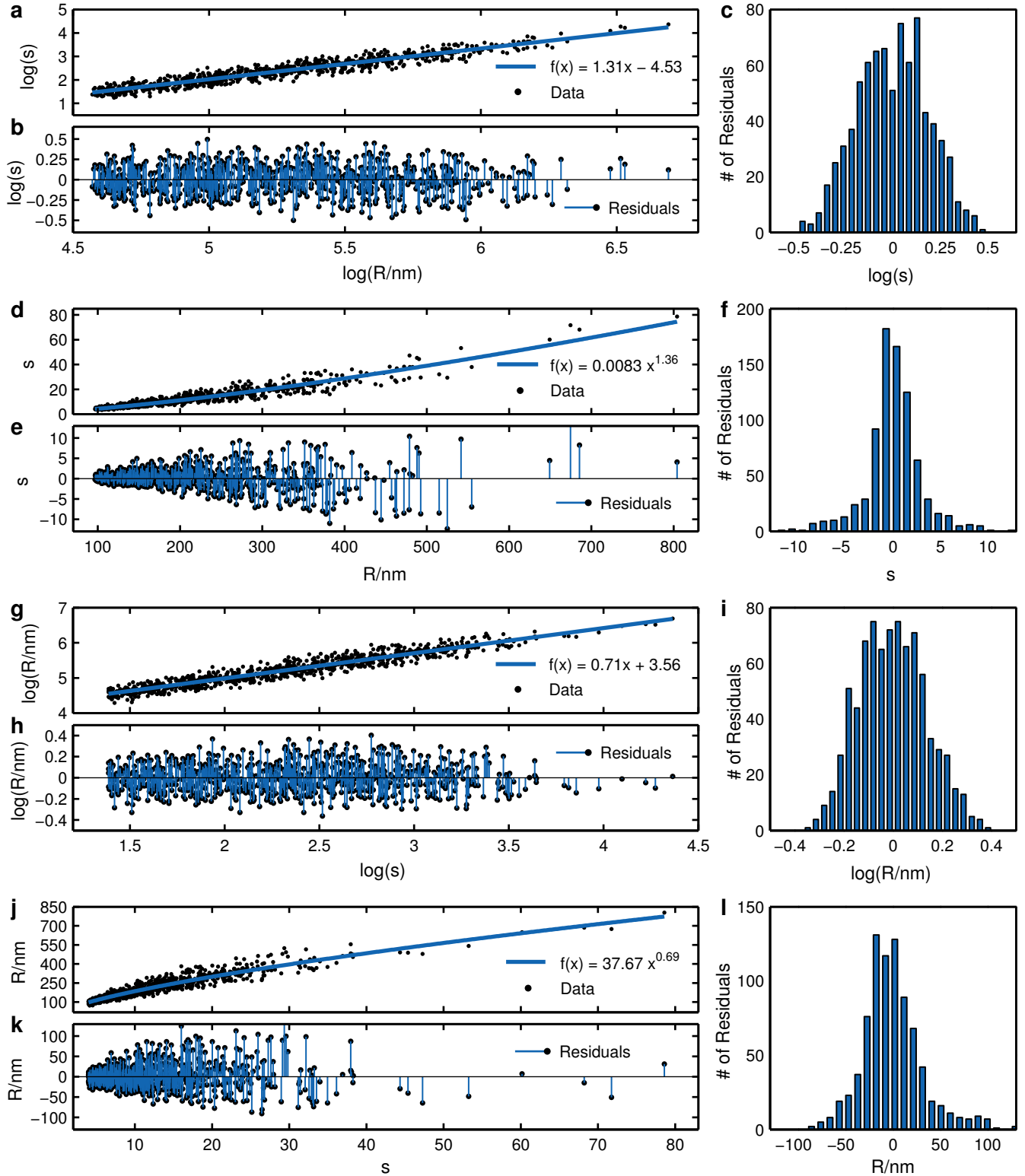


Figure S2: Residual analysis of different regressions aimed at calculating the fractal dimension  $D_f^{(R)}$  from cluster size  $s$  and radius of gyration  $R$  data. (a) and (g) show linear regressions (LR) on log-transformed data, while (d) and (j) show nonlinear regressions (NLR). (b), (e), (h), and (k) show the residuals of each corresponding fit, while (c), (f), (i), and (l) contain histograms showing the distributions of these residuals. The original data used for each of the four regressions are identical.

## Validation of PICT algorithms

In order to validate our image analysis software, we applied it to various SEM images and compared the results with those obtained manually or through the use of other software (such as the popular *ImageJ*). Some results from these validation studies are presented below.

### Validation of particle detection algorithms

The procedure used by PICT to detect particles is described in Section 2.3 of the main text. Particles are identified on the basis of three object properties—area, solidity, and eccentricity—whose acceptable ranges are specified by the user during calibration. Figures S3 and S4 show particles detected by PICT in a particular SEM image taken from the Ag nanocubes experiment (E1). The property ranges that were used in identifying cubes are listed in Table 1 in the main text. Note that not all single particles need to be detected, and that particles forming part of larger clusters may also be detected if they are separated from nearby particles by small gaps. This behavior is intentional.

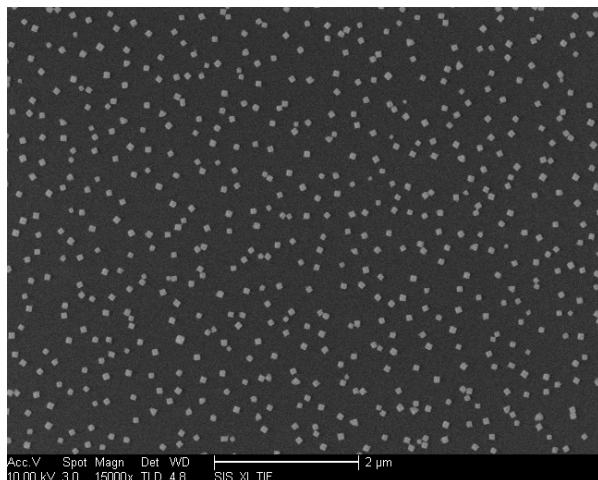


Figure S3: An SEM image taken at time point  $t = 135$  minutes in the experiment involving Ag nanocubes (E1).

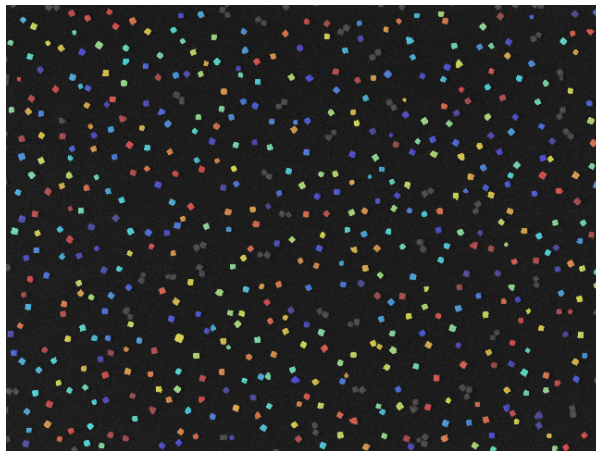


Figure S4: Cropped image with detected particles marked in color. PICT found 514 particles in this image.

### Validation of algorithms for identifying clusters

The procedure used by PICT to identify clusters is described in Section 2.3 of the main text, and involves a morphological closing operation with a small structuring element. This joins any particles that are not quite touching, yet are close enough to be considered part of a cluster. The user may specify the gap below which particles are joined in this manner (by specifying the radius  $r_S$  of the structuring element in nanometers). Different specifications can lead to a group of particles being identified either as a single cluster, or as multiple clusters. We have found that a good choice for  $r_S$  is half the distance between an average pair of NPs in a cluster (as determined beforehand by manual inspection and measurement).

Figures S5 and S6 show clusters identified by PICT in a particular SEM image taken from the Ag nanocubes experiment (E1), using a value of  $r_S = 42\text{nm}$ . Any border-touching

clusters are ignored, as discussed in Section 2.4 of the main text. Figures S7 and S8 show clusters identified in an image taken from the experiment involving 13nm Au nanospheres (E3). Note that the nanospheres are often separated by significant gaps even when they are part of a single cluster. To account for this, we used a value of  $r_S = 12\text{nm}$ , which is approximately twice the radius of a bare sphere.

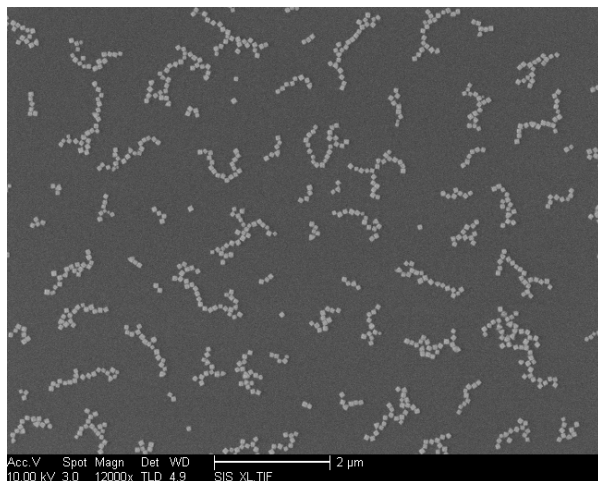


Figure S5: An SEM image taken at time point  $t = 180$  minutes in the experiment involving Ag nanocubes (E1).

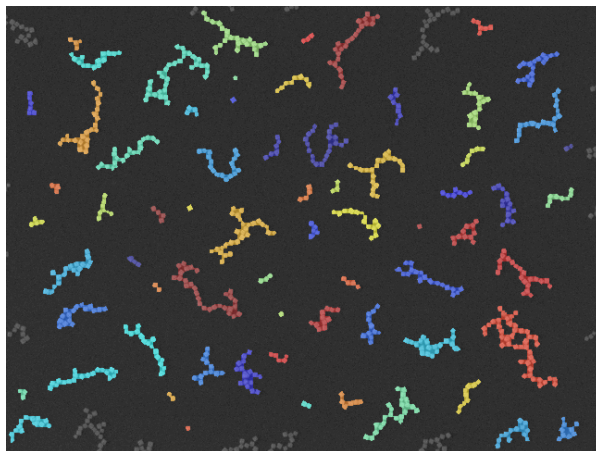


Figure S6: Cropped image with identified clusters marked in color. Each continuous colored region is one cluster.

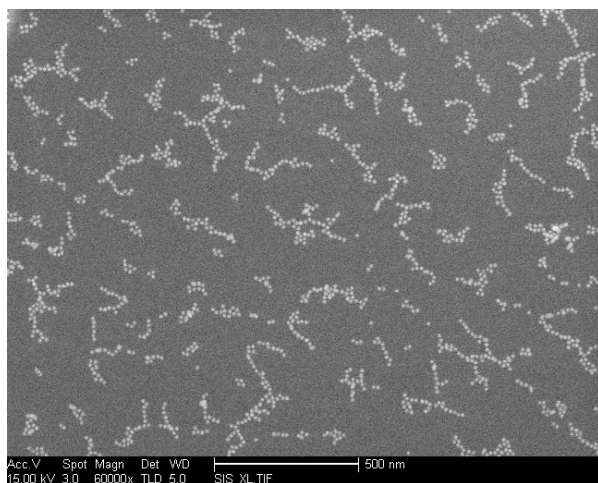


Figure S7: An SEM image taken at time point  $t = 85$  minutes in the experiment involving 13nm Au nanospheres (E3).

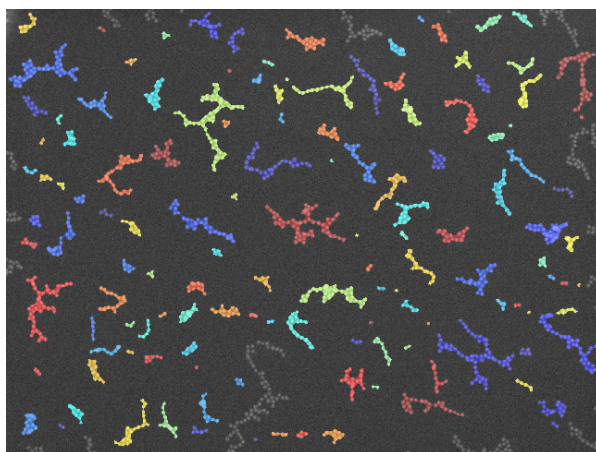


Figure S8: Cropped image with identified clusters marked in color. Each continuous colored region is one cluster.

### Validation of cluster property calculations

The methods used to calculate cluster properties are discussed in Sections 2.5-2.8 of the main text. In order to validate these methods, we compared the PICT results against

those obtained from visual inspection of the images and from analysis using the popular *ImageJ* software. Validation results for a particular SEM image taken from the Ag nanocubes experiment (E1) are shown below. Figure S9 is the original SEM image, while Figure S10 is a cropped and binary-transformed version that was imported into *ImageJ* and analyzed. Note that a few extra white pixels had to be added to this binary image so that *ImageJ* would correctly identify each cluster as a single object. All identified clusters are marked and those above a size threshold, numbered, in Figure S11. Figure S12 shows the calculated backbones and fitted ellipses for these clusters; these may be verified by visual inspection. Some other computed cluster properties are listed in Table S1 and compared with values obtained via hand-counting or from *ImageJ*.

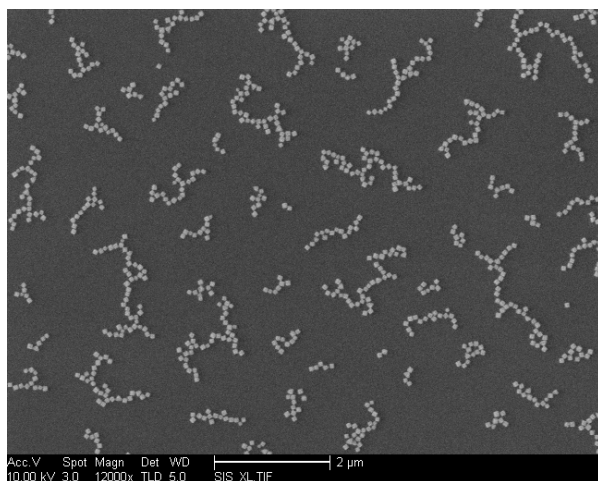


Figure S9: SEM image taken at time point  $t = 188$  minutes in the experiment involving Ag nanocubes (E1).

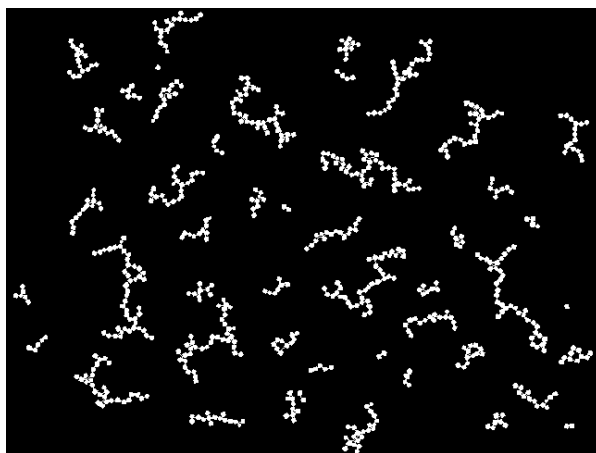


Figure S10: Cropped, binary-transformed image with border-touching clusters removed. This was analyzed with *ImageJ*.

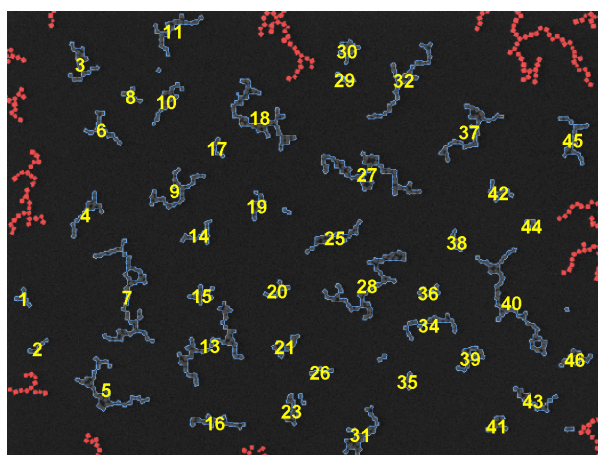


Figure S11: Valid clusters identified by PICT (48 in all) are marked in blue. Clusters of estimated size  $s \geq 4$  are numbered. Border-touching clusters are marked in red.

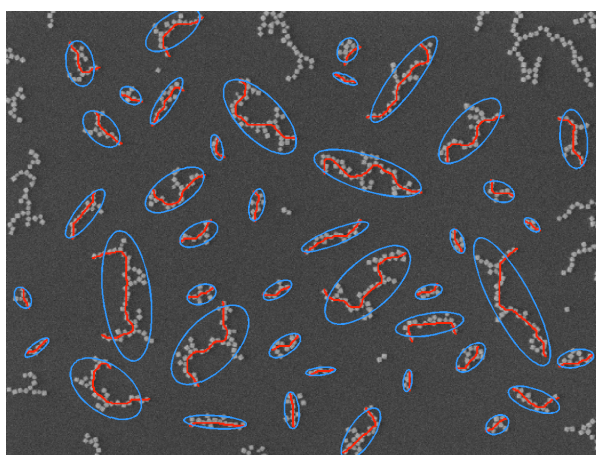


Figure S12: Cluster backbones (red) and fitted ellipses (blue), as determined by PICT, are shown for all clusters that are numbered in Figure S11.



Table S1: Comparison of PICT results with hand-counting (count) and ImageJ (ImJ) results for some basic cluster properties. The clusters are those that are numbered in Figure S11.  $s$  is the true cluster size, while  $\tilde{s}$  is an estimate computed using Eq. 6 in the main text. Note that  $\tilde{s}$  is not directly used by PICT in calculating size statistics.  $AR$  is the aspect ratio.

Cluster #	$\tilde{s}$ (PICT)	$s$ (count)	$AR$ (PICT)	$AR$ (ImJ)	Solidity (PICT)	Solidity (ImJ)
1	5.4	6	1.506	1.502	0.53	0.51
2	5.0	5	3.149	3.160	0.55	0.53
3	15.7	16	1.661	1.666	0.37	0.38
4	13.1	14	3.215	3.211	0.35	0.35
5	29.0	29	1.721	1.720	0.25	0.26
6	13.0	14	1.671	1.677	0.36	0.36
7	45.3	46	2.755	2.761	0.19	0.20
8	6.8	6	1.486	1.486	0.56	0.53
9	21.3	22	2.098	2.095	0.31	0.31
10	13.7	14	3.615	3.617	0.48	0.48
11	17.4	17	2.125	2.125	0.29	0.30
13	36.7	38	1.932	1.931	0.23	0.23
14	10.1	11	1.963	1.951	0.40	0.40
15	8.6	8	1.567	1.581	0.43	0.44
16	14.1	15	4.929	4.996	0.40	0.41
17	4.7	5	2.384	2.399	0.53	0.51
18	37.3	37	2.204	2.201	0.29	0.30
19	8.4	9	2.232	2.233	0.50	0.51
20	6.9	7	2.218	2.224	0.45	0.43
21	10.0	10	2.092	2.080	0.52	0.51
23	9.3	9	2.543	2.568	0.48	0.48
25	15.1	16	4.657	4.702	0.35	0.36
26	4.7	5	3.961	3.974	0.57	0.56
27	41.4	43	3.194	3.166	0.30	0.31
28	39.2	38	2.210	2.231	0.26	0.26
29	4.2	4	3.431	3.445	0.59	0.56
30	9.5	10	1.318	1.313	0.52	0.52
31	20.4	22	2.419	2.440	0.37	0.38
32	28.9	28	4.089	4.111	0.26	0.27
34	15.8	16	3.169	2.551	0.30	0.62
35	4.3	4	2.555	3.190	0.64	0.30
36	6.7	7	2.094	2.124	0.52	0.53
37	26.8	26	2.384	2.399	0.28	0.28
38	6.7	7	2.052	2.065	0.59	0.60
39	11.9	12	1.847	1.848	0.47	0.47
40	40.7	38	3.640	3.673	0.20	0.20
41	7.8	7	1.431	1.431	0.53	0.53
42	7.6	8	1.647	1.647	0.40	0.40
43	12.9	13	2.566	2.584	0.37	0.36
44	4.5	4	1.955	1.957	0.70	0.68
45	16.1	16	2.200	2.202	0.28	0.28
46	11.0	10	2.146	2.121	0.52	0.53

## Validation of Bayesian algorithm for computing cluster size statistics

The Bayesian algorithm used by PICT to calculate cluster size distributions is described in Section 2.5 of the main text. In order to validate this algorithm, we selected a subset of the SEM images from our Ag nanocubes experiment (E1); three images were chosen at each of the time points  $t = 127, 135, 143, 150, 158$  and  $165$  minutes. For these images, we manually identified all clusters and found their sizes (by counting the number of particles making up each cluster). This data was then used to compute the relative size distribution  $\nu_s(t)$  (see Eq. 3 in the main text). Figure S13 compares the manual result with the distributions calculated by PICT from the same set of images.

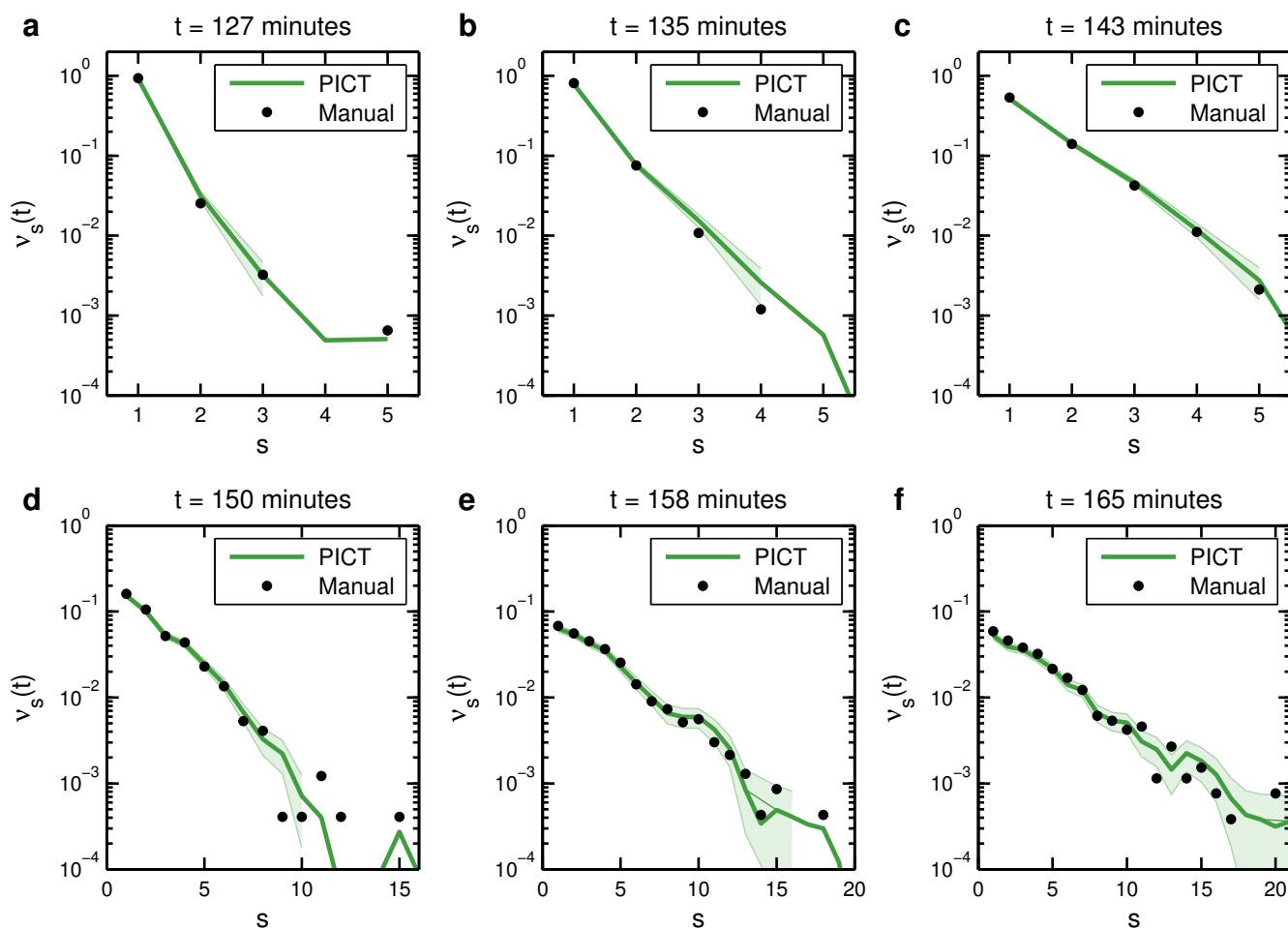


Figure S13: Comparison of relative size distributions  $\nu_s(t)$  calculated by PICT (green line) and manually (black dots) for a particular set of SEM images. Three images were analyzed at each time point; these images are from our Ag nanocubes experiment (E1). The lighter green regions are error bounds returned by PICT; these are not plotted for sizes where the predicted value is smaller than the error. Agreement between the results is in general excellent; slight deviations are due to the bias-correction algorithms used by PICT (see Section 2.4 in the main text), or simply due to statistical fluctuations (from lack of data) at large cluster sizes.

## References

- (S1) Xiao, X.; White, E. P.; Hooten, M. B.; Durham, S. L. On the use of log-transformation vs. nonlinear regression for analyzing biological power laws. *Ecology* **2011**, *92*, 1887–94.