Robust design of semi-automated clustering models for 4D-STEM datasets

Alexandra Bruefach
Department of Materials Science and Engineering
University of California, Berkeley, Berkeley, CA 94706
alexandra_bruefach@berkeley.edu

Colin Ophus
National Center for Electron Microscopy, Molecular Foundry, Lawrence Berkeley National Laboratory
1 Cyclotron Road, Berkeley, CA, USA, 94720
clophus@lbl.gov

M.C. Scott
Department of Materials Science and Engineering
University of California, Berkeley, Berkeley, CA 94706
National Center for Electron Microscopy, Molecular Foundry, Lawrence Berkeley National Laboratory
1 Cyclotron Road, Berkeley, CA, USA, 94720
mcscott@lbl.gov

Abstract

Materials discovery and design require characterizing material structures at the nanometer and sub-nanometer scale. Four-Dimensional Scanning Transmission Electron Microscopy (4D-STEM) resolves the crystal structure of materials, but many 4D-STEM data analysis pipelines are not suited for identification of anomalous and unexpected structures. This work introduces improvements to the iterative Non-Negative Matrix Factorization (NMF) method by implementing consensus clustering for ensemble learning. We evaluate the performance of models during parameter tuning and find that consensus clustering improves performance in all cases and is able to recover specific grains missed by the best performing model in the ensemble. The methods introduced in this work can be applied broadly to materials characterization datasets to aid in the design of new materials.

1 Introduction

Acceleration of materials discovery and design necessitates versatile and robust characterization tools that can resolve nm and sub-nm structures. Four-Dimensional Scanning Transmission Electron Microscopy (4D-STEM) is a technique that allows for structural maps to be created over micron field of view at high spatial resolution [1]. 4D-STEM datasets consist of a 2D diffraction pattern for each position in a 2D real space scan (Figure 1a). Crystallographic orientations [2–5], phases [6, 7], and properties [8–11] have been extracted from 4D-STEM datasets, but automated and semi-automated analysis pipelines have not been universally established for novel material systems [12].

Researchers have designed protocols for data analysis using template matching procedures [6, 13–16] and machine learning [17–20]. Template-based techniques can be useful when the phases present in the material are known and the classification problem is simple, but complex or anomalous structures often arise during the process of designing new materials. Deep learning approaches often depend on

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

simulated data for training, which may not reflect the complexity of new materials where structures or properties outside the training set may arise. Supervised methods cannot capture information that deviates from our current knowledge, which can prevent these methods from aiding in materials discovery and design pipelines. Thus, unsupervised learning is a practical alternative to rapidly identify regions of self-similarity within a dataset [21].

Unsupervised learning pipelines for 4D-STEM data analysis have been introduced in the past [5, 19, 22–24].These approaches primarily focus on either detecting unique Bragg reflections and using this form for clustering [5] or virtual imaging [22, 24]. Here, we present guidelines for the implementation of unsupervised learning and new approaches for improved performance. We first discuss the impact of parameter selection on performance to guide the implementation process for different feature sets. We then apply consensus clustering [21, 25, 26] to improve performance of our pipeline. The approaches we present can be extended broadly to other materials characterization techniques, where experts can design relevant featurization protocols [27].

2 Methods

We simulated 3 datasets containing Ag grains with different extents overlapping regions. The maximum diffraction pattern and true cluster labels for the datasets are shown in Figure 1b and 1c, respectively. We extracted features from these datasets using a mean virtual imaging method referred to as the Angular Average (AA) [24] and the detected Bragg Disk intensities and positions (BD) [5] using methods available in `py4DSTEM`[28]. We used a binning of 3x3 for the BD representation and an averaging step of 5 pixels for the AA representation. These choices maintained integrity of the individual disk positions and intensities while significantly reducing the size of the input feature set. These selections are not universally optimal and different binning and averaging steps are needed for datasets with different imaging conditions. Labels were generated from the simulated dataset for scoring purposes, but were not used during modeling. More details regarding our methods can be found in the Appendix.

Iterative NMF was introduced as a computational method in Allen et al. [5] and used in Bruefach et al. [24]. NMF ($||V - WH||_F, W \geq 0, H \geq 0$) reduces the feature matrix ($V, n \times m$) into a set of weighted linear combinations [29, 30]. We first apply NMF to the feature set, then merge the columns of the reduced component matrix ($W, n \times c$) that are correlated above the defined merge threshold. This is repeated until there are no components that are correlated above the merge threshold. The final columns in the component matrix become the clusters, with each pattern (n) having a weight associated with the clusters. This method mitigates the need to define the number of components, but the selection of the maximum components and merge threshold can greatly impact the performance of the model. We investigated the impact of the input parameters by running 25 models with different initialization conditions for each of four parameter sets (P1-P4, Table 1) for the AA and BD featurizations, leading to a total of 200 models trained per dataset.

Table 1: Parameter Sets for NMF Models

Feature	Parameter Set	Input Components	Correlation Threshold
AA	1	50	0.35
AA	2	50	0.40
AA	3	60	0.35
AA	4	60	0.40
BD	1	60	0.20
BD	2	60	0.25
BD	3	75	0.20
BD	4	75	0.25

P1 and P3 have lower merge thresholds than P2 and P4, and P1 and P2 lower number of initial components than P3 and P4. The initial number of input components controls the maximum number of clusters that can be output from the model. If the value of initial components is too low, there may be some grains that do not get detected. While a value too large may not directly impact the quality of results, it does cause runtimes that are unnecessarily long. The merge threshold controls how correlated similar clusters must be in order to be merged into one, thus a value larger than or equal to 1 would reduce the Iterative method to traditional NMF. If the selected merge threshold is too large (but still less than 1), patterns with the same underlying grain identity may be split into two clusters. Conversely, if this threshold is too low, dissimilar grains may be merged into one cluster.

Consensus clustering is performed by consolidating the clusters discovered from the 25 models per parameter set into one model. Unlike in supervised learning, the cluster assignment does not represent a class label and the number of clusters is not consistent among all models. To address these complications, we performed label correspondence to determine which clusters should be combined for the consensus. We selected the model with the maximum number of clusters out of the 25 trained models and used these as the foundation set. The remaining clusters were matched to the foundation set (A) using the sum of the weights in the intersection of the current cluster (B_j) and the foundational cluster (A_i) in the initialized bins (i) based on the equation $\mathrm{bin(B_j)} = \mathrm{argmax}(\sum_{\mathrm{i=1}}^{\mathrm{i=n}}(\mathrm{A_i} \bigcap \mathrm{B_j})$. We created a new bin if no match was found. All bins that contained less than two clusters were dropped. After matching the clusters, the average was taken in each bin and used as the consensus. We calculated the True Positive Rate (TPR), False Positive Rate (FPR), True Negative Rate (TNR), False Negative Rate (FNR) and F1 Score for the independent and consensus cluster models.

3 Results and Discussion

While many models successfully segmented the data (Figure 1d), we observed three distinct types of model failures (Figure 1e). The first is the inability of the model to retain information, leading to low F1 scores and high NMF reconstruction error. This type of error is easy to detect both computationally and visually (Figure 1e, middle row). Failure to retain information is likely due to poor initialization conditions and typically doesn't impact an entire parameter set, except for the P3 models using the BD feature for the Ag2 dataset. The AA feature also had several models per parameter set that failed in this way. The second type of failure can be observed from the Ag1 BD model results shown in Figure 1e (top row). This is associated with too high of a merge threshold, which leads to multiple clusters representing single grains instead of one cluster. These two failure cases underpin the necessity to test parameter sets for specific datasets when performing Iterative NMF as the primary clustering model. The last failure case is the inability to detect the overlapping grains as two distinct clusters, shown in Figure 1e, (bottom row). We observe this failure primarily in models using the BD feature. Previous work has shown that the BD feature has the tendency to overfit the overlapping regions by clustering them into separate classes rather than containing two distinct grains [24]. In addition to these failures, none of the models detect all of the grains in the datasets. This can be observed in the Ag1 cluster map in Figure 1d (top row), where the model omits some grains present in the Figure 1c true cluster maps.

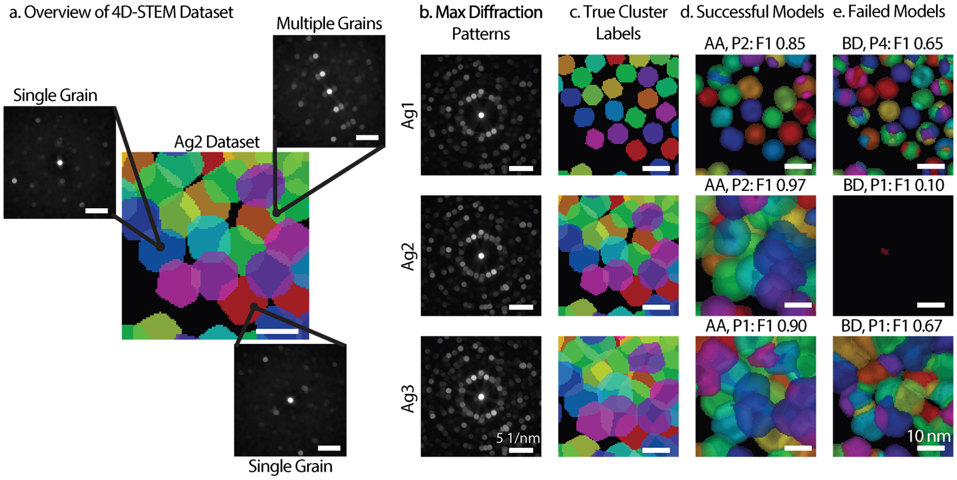

Figure 1: Overview of 4D-STEM with visual representation of datasets and results. (a) Diagram of the 4D-STEM dataset. (b) Maximum diffraction patterns and (c) true cluster labels for the 3 simulated datasets, Ag1 (top), Ag2 (middle) and Ag3 (bottom). (d) Example of a successful model for each dataset, (e) example of a failed model for each dataset.

The F1 scores for the models trained for Ag1, Ag2, and Ag3 are shown in Figure 2a, 2b, and 2c, respectively. Additional performance scores are presented in Table S1. We found that the model set using P1 performs well across all 3 datasets using the BD representation. We believe the success of P1 changes for different datasets. First, the lower merge threshold leads to improved performance for Ag1 by preventing independent grains from being put into separate clusters. In Ag3, the enhanced performance from P1 may be due to the lower number of components. All parameter sets applied to Ag2 overfit to the overlapping grain regions, but the lower number of components may prevent some of the overlapping regions from being clustered as distinct from the parent grains. The lower number of components initializes fewer clusters, thus this could reduce the number of overlapping regions that are initially identified as an independent cluster. We find that even when datasets are very similar, parameter tuning is a crucial step for optimizing independent model performance.

The models developed using the AA feature with P4 perform consistently well across all datasets. We believe that the high number of components for the AA feature is important for Ag1, which may provide a higher possibility of initializing cluster for more of the independent grains in the dataset. The higher merge threshold in this set leads to high performance of P4 on the Ag2 and Ag3 datasets, which prevents dissimilar grains from being merged into a single cluster. It is possible that the lower merge threshold tends to merge dissimilar grains when applying to the AA feature when there is some extent of grain overlap. If not much is known about the content of the dataset, using AA feature with P4 would be a good choice as these models have the best overall performance across the datasets, with an average overall F1 score of 0.84 across the 75 models and 3 datasets (Table S2).

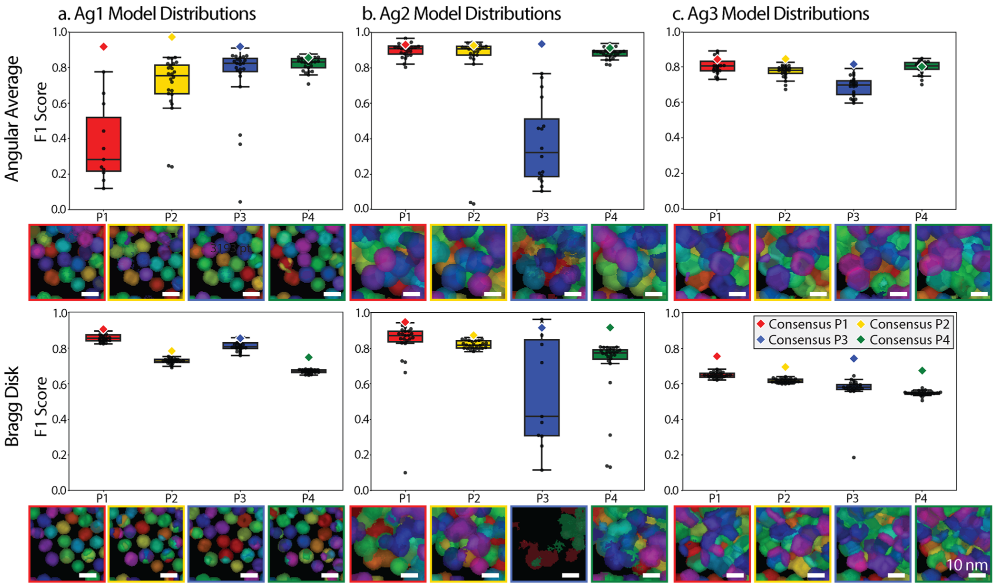

Figure 2: Comparison of consensus cluster and F1 scores. Measurements and consensus cluster maps for Ag1 (a), Ag2 (b) and Ag3 (c) trained using 4 different parameter sets for the AA (top) and BD (bottom) representations.

We observed that the Ag2 dataset is more sensitive to input parameters for the BD feature. All sets using the BD feature for the Ag1 and Ag3 datasets have narrow distributions (average variance 0.001), yet the models using the BD feature for Ag2 tend to have broader distributions (average variance 0.04). We believe the small regions of overlapping grains is the cause of the variability. Since these regions are small relative to the grain size, only a few models per parameter set may see these regions as distinct. This theory is underpinned by the reproducible poor performance of all of the models created using the BD feature for Ag3 ($F1_{best}$ = 0.68). The Iterative NMF method may have difficultly determining minority phases if they are only present in few patterns per dataset, especially if there is significant overlap with majority crystallographic features in the sample. The BD feature is more sensitive to detecting these regions as they become larger. This may be beneficial or detrimental based on the classification problem. For example, the BD representation has a tendency to fail at identifying overlapping regions correctly, but for the same reason is more likely to properly identify distinct precipitates in a matrix or minority phases in a material system.

Next, we compared how consensus clustering performed against individual models in each parameter set. In all cases, the consensus cluster F1 score ($F1_{consensus}$) (Figure 2, diamond markers) improves above the average of the parameter set ($F1_{avg}$) and in several cases beats the best model ($F1_{best}$) within a parameter set. The largest performance enhancement occurs in sets containing some models that failed to retain information. However, consensus clustering did not improve clustering in parameter sets that did not retain any information, such as P3 using the BD feature for the Ag2 dataset. In addition to the benefit of improved performance, we observe from the cluster maps that information from individual grains that top performing models omit can be recovered using consensus clustering. For example, the top model for Ag1 using the AA feature in Figure 1b omits some grains, but these are now observed in the consensus cluster map for this parameter set. Thus, more grains can be accurately detected when taking the consensus over the best performing model.

Table 2: Average, best, and consensus F1 Scores in select parameter sets

Value	Ag1, AA	Ag2, AA	Ag3, AA	Ag1, BD	Ag2, BD	Ag3, BD
	P1	P3	P3	P4	P4	P3
$F1_{avg}$	0.37	0.38	0.69	0.67	0.70	0.57
$F1_{best}$	0.78	0.77	0.80	0.68	0.81	0.64
$F1_{consensus}$	0.92	0.93	0.82	0.75	0.92	0.74

Finally, the consensus clustering approach mitigates the impact of parameter selection on performance. The performance of the consensuses are relatively stable across parameter sets, as the $F1_{consensus}$ has a narrower distribution than both the $F1_{avg}$ and $F1_{best}$. However, the consensus clustering protocol applied in this work does not address the overfitting problem plaguing the models trained on the BD representation. Future work could address this shortcoming by incorporating merging and splitting rules to the consensus method.

4 Conclusion

Unsupervised data analysis methods for materials characterization are crucial for advancement of materials discovery and design efforts. This work introduces the nuances of parameter selection and performance variability in Iterative NMF applied to 4D-STEM datasets using engineered features as model inputs. We find datasets that have equally sized grains are less sensitive to input parameters. The models tend to have more difficulty resolving minority regions within a dataset, as indicated by the high model variance in the Ag2 dataset. While there is no universally superior feature set or parameter set for all material systems, the AA feature with P4 had the best average performance across our test sets. In general, we find the AA representation performs better when the number of unique structures per pattern increases, but the BD feature would be a better choice when attempting to identify minority phases. Consensus clustering consistently reduces error and recovers information that is lost in individual cluster sets, and reduces the impact of parameter selection on performance. We anticipate that these findings and methods will be applied to other materials characterization datasets to further support materials discovery efforts.

Acknowledgements

AB and py4DSTEM development are supported by the Toyota Research Institute. Work at the Molecular Foundry was supported by the Office of Science, Office of Basic Energy Sciences, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

Data Availability

The data and methods that support the findings presented in this work are available in Demonstration of Consensus Clustering for 4D-STEM at https://doi.org/10.5281/zenodo.7195135 [31].

References

[1] C. Ophus, *Microsc. Microanal.* **2019**, *25*, 563–582.

[2] E. D. Grimley, S. Frisone, T. Schenk, M. H. Park, C. M. Fancher, T. Mikolajick, J. L. Jones, U. Schroeder, J. M. LeBeau, *Microsc. Microanal. Conference Proceedings* **2018**, *24*.

[3] D. Mukherjee, J. T. L. Gamler, S. E. Skrabalak, R. R. Unocic, *ACS Catalysis* **2020**, *10*, 5529–5541.

[4] A. Londono-Calderon, D. J. Williams, M. M. Schneider, B. H. Savitzsky, C. Ophus, S. Ma, H. Zhu, M. T. Pettes, *Nanoscale* **2021**, *13*, 9606–9614.

[5] F. I. Allen, T. C. Pekin, A. Persaud, S. J. Rozeveld, G. F. Meyers, J. Ciston, C. Ophus, A. M. Minor, *Microsc. Microanal.* **2021**, *1*, 1–10.

[6] A. K. Shukla, Q. M. Ramasse, C. Ophus, D. M. Kepaptsoglou, F. S. Hage, C. Gammer, C. Bowling, P. A. H. Gallegos, S. Venkatachalam, *Energy Environ. Sci.* **2018**, *11*, 830–840.

[7] E. F. Rauch, M. Veron, *Acta Crystallogr. Sect. B: Struct. Sci. Cryst. Eng. Mater.* **2019**, *75*, 505–511.

[8] T. C. Pekin, J. Ding, C. Gammer, B. Ozdol, C. Ophus, M. Asta, R. O. Ritchie, A. M. Minor, *Nat. Commun.* **2019**, *10*, 1–7.

[9] X. Mu, L. Chen, R. Mikut, H. Hahn, C. Kubel, *Acta Materialia* **2021**, *212*, 116932.

[10] N. Yang, C. Ophus, B. H. Savitzsky, M. C. Scott, K. Bustillo, K. Lu, *Mater. Charact.* **2021**, *181*, 1111512.

[11] E. Thornsen, J. Frafjord, J. Friis, C. Marioara, S. Wenner, S. Andersen, R. Holmestad, *Mater. Charact.* **2021**.

[12] A. Ponce, J. A. Aguilar, J. Tate, M. Jose Yacaman, *Nanoscale Adv.* **2021**, *3*, 311–325.

[13] E. R. Rauch, J. Portillo, S. Nicolopoulos, D. Bultreys, S. Rouvimov, P. Moeck, *Z Kristallogr Cryst Mater* **2010**, *225*, 103–109.

[14] T. Meng, J.-M. Zuo, *The European Physical Journal Applied Physics* **2017**, *80*, 107901.

[15] C. Ophus, S. E. Zeltmann, A. Bruefach, A. Rakowski, B. H. Savitzsky, A. M. Minor, M. Scott, *Microsc. Microanal.* **2021**, 1–14.

[16] N. Cautaerts, P. Crout, H. W. Anes, E. Prestat, J. Jeong, G. Dehm, C. H. Liebscher, *Ultramicroscopy* **2022**.

[17] B. H. Martineau, D. J. Johnstone, A. T. van Helvoort, P. A. Midgley, A. S. Eggeman, *Adv Struct Chem Imag* **2019**, *5*, 1–14.

[18] R. Yuan, J. Zhang, L. He, J.-M. Zuo, *Ultramicroscopy* **2021**, *231*, 113256.

[19] C. Shi, M. Cao, S. M. Rehn, S.-H. Bae, J. Kim, M. R. Jones, D. A. Muller, Y. Han, *npj Computational Materials* **2022**, *8*, 1–9.

[20] J. Munshi, A. Rakowski, B. H. Savitsky, S. E. Zeltmann, J. Ciston, M. Henderson, S. Cholia, A. M. Minor, M. K. Chan, C. Ophus, *(Preprint) arXiv:2202.00204* **2022**.

[21] H. G. Ayad, M. S. Kamel, *Pattern Recognition* **2010**, *43*, 1943–1953.

[22] A. Nalin Mehta, N. Gauquelin, M. Nord, A. Orekhov, H. Bender, D. Cerbu, J. Verbeeck, W. Vandervorst, *Nanotechnology* **2020**, *31*, 445702.

[23] F. Uesugi, S. Koshiya, J. Kikkawa, T. Nagai, K. Mitsuishi, K. Kimoto, *Ultramicroscopy* **2021**, *221*, 113168.

[24] A. Bruefach, C. Ophus, M. C. Scott, *Microscopy and Microanalysis* **2022**, 1–11.

[25] A. Strehl, J. Ghosh, *Journal of machine learning research* **2002**, *3*, 583–617.

[26] T. Boongoen, N. Iam-On, *Computer Science Review* **2018**, *28*, 1–25.

[27] S. K. Thati, J. Ding, D. Zhang, X. H. Hu, *IEEE Int. Conf. Softw. Qual. Reliab. Secur. QRS 2017* **2015**.

[28] B. H. Savitzky, S. E. Zeltmann, L. A. Hughes, H. G. Brown, S. Zhao, P. M. Pelz, T. C. Pekin, E. S. Barnard, L. Rangel DaCosta, E. Kennedy, Y. Xie, M. T. Janish, M. M. Schneider, P. Herring, C. Gopal, A. Anapolsky, R. Dhall, K. C. Bustillo, P. Ercius, M. C. Scott, H. Ciston, A. M. Minor, C. Ophus, *Microsc. Microanal.* **2021**, *27*, 1–32.

[29] P. Paatero, U. Tapper, *Environmetrics* **1994**, *5*, 111–126.

[30] D. D. Lee, H. S. Seung, *Nature* **1999**, *401*, 788–791.

[31] A. Bruefach, **2022**, DOI `https://doi.org/10.5281/zenodo.7195135`.

[32] J. M. Cowley, A. F. Moodie, *Acta Crystallographica* **1957**, *10*, 609–619.

[33] E. J. Kirkland, *Advanced computing in electron microscopy, 3rd edition*, Springer Science & Business Media, **2020**.

[34] C. Ophus, *Advanced structural and chemical imaging* **2017**, *3*, 1–11.

[35] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.

[36] S. van der Walt, J. L. Schonberger, J. Nunez-Iglesias, F. Boulgne, J. D. Warner, N. Yager, E. Gouilart, T. Yu, the scikit-image contributors, *PeerJ* **2014**, *2*, e453.

A Appendix

A.1 Expanded Methods

A.1.1 Dataset generation and feature preparation

Three simulated polycrystalline Ag films (referred to as Ag1, Ag2, Ag3) with varying grain sizes ((35 Å x 41 Å x 40 Å), (52 Å x 62 Å x 60 Å), (70 Å x 82 Å x 80 Å)) were generated using a custom MATLAB code. The 4D-STEM simulations were performed using custom MATLAB scripts that implement the multislice algorithm Cowley and Moodie [32] and methods defined by Kirkland [33] and the plane wave reciprocal space interpolated scattering matrix (PRISM) algorithm[34]. The patterns were generated using an acceleration potential of 300 keV, a probe semiconvergence angle of 1.05 mrad, a 5 Å pixel size in real space, and a 0.01 Å^{-1} pixel size in reciprocal space. More details about the dataset generation and simulation can be found in Bruefach et al. [24].

Rigorous data calibration steps have been introduced by Ophus [1] and Savitzky et al. [28]. In this work, we follow the pattern alignment and elliptical distortion correction methods using the publicly available methods in `py4DSTEM` [28]. We then extract the Angular Average (AA) and Bragg Disk (BD) representations following the methods discussed in Bruefach et al. [24]. A binning of 3x3 for the BD representation and an averaging of 5 pixels for the AA representation were chosen because they maintained integrity of the individual disk positions and intensities while reducing the size of the input data. These selections are not universally optimal and different parameters may be a better fit for datasets with different imaging conditions.

A.1.2 Compute Details

Iterative NMF was performed as discussed in the main text, using the implementation of NMF in `scikit-learn`[35] at each iteration. Runtime was measured using a MacBook Pro equipped with a 2.3 GHz 8-Core Intel Core i9 processor, and ranged from 65 minutes to 270 minutes. The implementation of the Iterative NMF is the compute bottleneck for this pipeline, all other data preparation and analysis methods take minimal time. Typically, smaller number of components and larger merge thresholds yield smaller compute times. This implementation is currently only implemented for use on a CPU and we expect significant speedups upon implementing for GPU. More specific details are presented in a prior study [24].

A.1.3 Spatial Separation and Size Refinement

We spatially separated the components and filtered out small grains using the real space cluster maps in order to score the datasets. We transformed each column of the NMF output components into 2D arrays of the size of the real space scans (100 x 100 in the simulated dataset). The yen threshold function from `scikit-image` [36] was applied to each image produced from the NMF output, which allowed for the identification of spatially independent clusters within the cluster image. All values below the threshold in each image were set to 0. Any spatially independent components containing less than 25 pixels were removed. The clusters were separated based on the detected labelled region. This method can be applied to experimental data if the analysis of the size, diameter, and shape of detected clusters is desired. However, applying spatial separation would prevent spatially separated clusters of similar orientation from maintaining the same cluster label.

A.1.4 Scoring

The Ag labels for the three Ag datasets were generated based on the Ag structures at each probe position. To determine the best match grain, the sum of the weights in the intersect of the grain and cluster was calculated. The best match was After the best match was found and the True Positive Rate (TPR), True Negative Rate (TNR), False Positive Rate (FPR), False Negative Rate (FNR), and F1 Score were calculated for each pair within a cluster. The average of each of these parameters was calculated for each model and variations within and across parameter sets were evaluated. In this work, we use the F1 score to gauge model performance, but all values are tabulated in the SI.

A.2 Tables

Table S1: Average metrics, per parameter set

Dataset	Feature	Parameter Set	TPR	FPR	TNR	FNR	F1 Score
Ag1	AA	P1	0.27	0.00	1.00	0.73	0.37
Ag1	AA	P2	0.64	0.00	1.00	0.36	0.70
Ag1	AA	P3	0.70	0.01	0.99	0.30	0.75
Ag1	AA	P4	0.79	0.01	0.99	0.21	0.82
Ag1	BD	P1	0.80	0.00	1.00	0.20	0.85
Ag1	BD	P2	0.62	0.00	1.00	0.38	0.72
Ag1	BD	P3	0.74	0.00	1.00	0.25	0.81
Ag1	BD	P4	0.55	0.00	1.00	0.45	0.67
Ag2	AA	P1	0.87	0.01	0.99	0.13	0.90
Ag2	AA	P2	0.81	0.02	0.98	0.19	0.83
Ag2	AA	P3	0.28	0.00	1.00	0.72	0.38
Ag2	AA	P4	0.86	0.02	0.98	0.14	0.88
Ag2	BD	P1	0.77	0.02	0.98	0.23	0.82
Ag2	BD	P2	0.73	0.01	0.99	0.26	0.82
Ag2	BD	P3	0.47	0.01	0.99	0.53	0.55
Ag2	BD	P4	0.60	0.01	0.99	0.40	0.70
Ag3	AA	P1	0.76	0.01	0.99	0.24	0.81
Ag3	AA	P2	0.70	0.00	1.00	0.30	0.78
Ag3	AA	P3	0.62	0.00	1.00	0.38	0.69
Ag3	AA	P4	0.75	0.01	0.99	0.25	0.80
Ag3	BD	P1	0.52	0.00	1.00	0.48	0.65
Ag3	BD	P2	0.48	0.00	1.00	0.52	0.62
Ag3	BD	P3	0.44	0.00	1.00	0.56	0.57
Ag3	BD	P4	0.40	0.00	1.00	0.60	0.55

Table S2: Averaged model metrics across datasets

Feature	Parameter Set	TPR	FPR	TNR	FNR	F1 Score
AA	P1	0.64	0.01	0.99	0.36	0.69
AA	P2	0.72	0.01	0.99	0.28	0.77
AA	P3	0.54	0.00	1.00	0.46	0.61
AA	P4	0.80	0.01	0.99	0.20	0.84
BD	P1	0.70	0.01	0.99	0.30	0.77
BD	P2	0.61	0.00	1.00	0.39	0.72
BD	P3	0.55	0.01	0.99	0.45	0.64
BD	P4	0.52	0.00	1.00	0.48	0.64
AA	all	0.67	0.01	0.99	0.33	0.73
BD	all	0.59	0.00	1.00	0.41	0.69

Table S3: Average metrics, Consensus

Dataset	Feature	Parameter Set	TPR	FPR	TNR	FNR	F1 Score
Ag1	AA	P1	0.91	0.02	0.98	0.09	0.92
Ag1	AA	P2	0.97	0.02	0.98	0.03	0.97
Ag1	AA	P3	0.91	0.02	0.92	0.09	0.92
Ag1	AA	P4	0.83	0.01	0.99	0.17	0.86
Ag1	BD	P1	0.87	0.00	1.00	0.13	0.90
Ag1	BD	P2	0.69	0.00	1.00	0.31	0.78
Ag1	BD	P3	0.81	0.00	1.00	0.19	0.85
Ag1	BD	P4	0.65	0.00	1.00	0.35	0.75
Ag2	AA	P1	0.93	0.04	0.96	0.07	0.93
Ag2	AA	P2	0.93	0.04	0.96	0.07	0.93
Ag2	AA	P3	0.92	0.03	0.97	0.08	0.93
Ag2	AA	P4	0.91	0.04	0.96	0.09	0.91
Ag2	BD	P1	0.95	0.05	0.95	0.05	0.95
Ag2	BD	P2	0.82	0.01	0.99	0.18	0.87
Ag2	BD	P3	0.98	0.20	0.80	0.02	0.92
Ag2	BD	P4	0.89	0.03	0.97	0.11	0.92
Ag3	AA	P1	0.83	0.02	0.98	0.17	0.85
Ag3	AA	P2	0.81	0.01	0.99	0.19	0.85
Ag3	AA	P3	0.77	0.02	0.98	0.23	0.83
Ag3	AA	P4	0.77	0.02	0.98	0.23	0.81
Ag3	BD	P1	0.65	0.01	0.99	0.35	0.75
Ag3	BD	P2	0.58	0.00	1.00	0.42	0.69
Ag3	BD	P3	0.64	0.01	0.99	0.36	0.74
Ag3	BD	P4	0.55	0.00	1.00	0.45	0.67