Semi-automated segmentation of atomic-resolution microscopy images with unsupervised machine learning

Ning Wang, Christoph Freysoldt, and Christian Liebscher

Max-Planck-Institut für Eisenforschung GmbH, Max-Planck-Straße 1, 40237 Düsseldorf, Germany

We present an unsupervised machine learning approach for semi-automated segmentation of atomic-resolution microscopy images. We combine our newly-proposed symmetry-based local descriptors with the classical unsupervised machine learning algorithms, and highlight that the microscopy images might be segmented in an unsupervised manner bypassing the tedious training of supervised machine learning models. Our approach has general applicability for images with atomic resolution, and in this paper, we demonstrated the successful application to the high-angle annular dark-field scanning Transmission electron microscopy (HAADF-STEM) images. We released our code as a python module that reads the microscopy images and outputs the labels for all pixels, which is flexible to be used either as a standalone python code or as a plugin to other microscopy packages.

I. INTRODUCTION

The development of materials science is largely driven by that of microscopes. Since the second half of the last century, the substantial advances of microscopy have made it possible to probe materials with atomic resolution [1], which opens up an era where the materials can be studied down to the level of singe atoms. Accompanying the ubiquitous acquisition of data at the atomic scale, we are facing the challenge of efficiently analyzing the large amount of data generated even in a single experiment [2]. The machine-learning methods have been shown to be effective in speeding up and automatizing the data processing in various microscopy-related problems [3–5].

The high-resolution imaging microscopes, such as STEM, have made it routine to acquire atomic-resolution images and videos. To analyze the large volume of microscopy images becomes a tedious task for researchers, and the manual imaging analysis has the drawback of inconsistency in interpreting the image either qualitatively or quantitatively. Among various image analysis tasks, the image segmentation is crucial as it relates the digital signals to crystal patterns and serve as a necessary step for further analysis, and challenging as each pixel in the image needs to be interpreted correctly according to the features of its local surroundings and linked to the correct pattern label. The recent substantial advances of machine learning and computer vision provides powerful tools to resolve this challenge and aid us to develop the efficient and robust approach for the segmentation of atomic-resolution microscopy images, which is the focus of this paper.

As the difference between the crystal patterns in the atomic-resolution images are dominated by the arrangement of atoms, the conventional segmentation methods such as thresholding [6, 7] and region merging [8, 9] are not straightforwardly applicable for this task. The supervised machine learning methods [10, 11] have been explored in the community of biology to segment the tissue images, which require a large amount of labeled data to train a classification model or a neural network. While the good progress has been demonstrated in literature, the drawback of the supervised models are also transparent. The data labeling is time-consuming and the supervised model has no general applicability and only works for the images that are similar to those in training data. Our approach is based on the consideration that the segmentation of atomic-resolution images is mostly based on the local symmetry, which might be encoded into the local descriptors. The local descriptors can then be fed into the unsupervised clustering algorithms in order to segment the microscopy images. The closest work to ours is probably the machine learning-based workflow for segmentation of atomic force microscopy (AFM) images that was published by Borodinov et al.[12] very recently. In their workflow, the local descriptors are first calculated with a pre-selected transform (Discrete Fourier, wavelet, or Radon transform), which are fed into the clustering algorithms to segment the AFM images. Our approach shares the similar architecture with their workflow. What makes our work different stem from several aspects. We exploit the self-similarity inherent in the microscopy images and propose the symmetry-based local descriptors. We find that an up-sampling plus stride scheme effective to speed up the segmentation. We further package our code as a python module, which is ready to use simply by typing several lines of python code and flexible to be plugged into other microscopy packages.

Our segmentation approach consists of four steps:
1. calculate the symmetry-based local descriptors;
2 perform principle component analysis to reduce the dimension;
3. perform K-means clustering in the descriptor space;
4. perform up-sampling to match the shape of the original image.

There are several hyperparameters to be specified by the users, and this is the reason why we claim our approach to be semi-automated. Once the hyperparameters are specified, all of the rest are performed by the computer automatically.

Next, we explain each step of our approach in the methodology section, and in the application section, we

present two examples of the application to HAADF-STEM images.

II.   METHODOLOGY

A.   Feature extraction via symmetry operations

We aim at segmenting the microscopy images into multiple crystal patterns that are different in terms of symmetry, i.e., the partitions are invariant with respect to different symmetry operations. Four types of symmetry operations are involved in the two-dimensional crystal patterns, translations, rotations, reflections, and glide-reflections, and they are dictated by the relations below,

$$
\begin{aligned}
\text{translation} &: \mathbf{r}' = \mathbf{r} + \mathbf{t}, \\
\text{rotation} &: \mathbf{r}' = \mathbf{R}\mathbf{r}, \\
\text{reflection} &: \mathbf{r}' = \mathbf{M}\mathbf{r}, \\
\text{glide} - \text{reflection} &: \mathbf{r}' = \mathbf{M}\mathbf{r} + \mathbf{t},
\end{aligned}
\tag{1}
$$

where r and $\mathbf{r}'$ are the positions of the original and the transformed pixels, $\mathbf{t}$ the translation vector, $\mathbf{R}$ the rotation matrix, and $\mathbf{M}$ the reflection matrix. As the entries in translation vector and the rotation and reflection matrices may take various values, there are many possible symmetry operations. We choose one from them, and denote it as the candidate symmetry operation. If

we apply it to a crystal pattern and score the similarity between the original and the transformed ones, the score would be high if it is the symmetry operation of the examined the pattern and low if it is not. The scoring of a series of candidate symmetry operations gives us a symmetry-score vector, which can be used as a descriptor to characterize the symmetry of the pattern. In the segmentation task, since we need to examine the local symmetry in the microscopy image, we simply apply the symmetry operations to the patches instead of the whole image, and then use the symmetry-score vector as the local descriptor. In other words, the candidate symmetry operation together with the symmetry scoring makes a local-symmetry-feature extractor. We shift the extractor across the whole image in order to extract the local-symmetry information everywhere in the image. The schematic diagram for this calculation is shown in Fig. 1.

We expect the symmetry score to measure the similarity between the original and the transformed patches. The Pearson's correlation coefficient is a good choice for this purpose, which takes a value in the range from -1 to 1 and and a larger value for the stronger similarity. For the original patch $P_{x,y}$ centered at the pixel (x, y), and the corresponding transformed patch $OP_{x,y}$ by the operation O, the cross-correlation coefficient is evaluated according to

$$
\begin{aligned}
\mathrm{SC}_O(x, y) &\equiv \mathrm{Corr}\left(P_{x,y}, OP_{x,y}\right) \\
&= \frac{\sum_{i=1}^{p_x}\sum_{j=1}^{p_y}[P_{x,y}(i,j) - \mu_{P_{x,y}}] \cdot OP_{x,y}(i,j) - \mu_{OP_{x,y}}]}{\sqrt{\sum_{i=1}^{p_x}\sum_{j=1}^{p_y}[P_{x,y}(i,j) - \mu_{P_{x,y}}]^2} \cdot \sqrt{\sum_{i=1}^{p_x}\sum_{j=1}^{p_y}[OP_{x,y}(i,j) - \mu_{OP_{x,y}}]^2}},
\end{aligned}
\tag{2}
$$

where O is an arbitrary symmetry operation, $P_{x,y}$ the original patch centered at (x, y), $OP_{x,y}$ the transformed patch. Both patches have a size of (p_x, p_y). $P_{x,y}(i, j)$ and $P_{x,y}(i, j)$ are the intensities at the position (i, j) in their relative coordinates. $\mu_{P_{x,y}}$ and $\mu_{OP_{x,y}}$ are the mean intensities of the patches $P_{x,y}$ and $OP_{x,y}$. We emphasize that the relative coordinate is not necessarily the Cartesian coordinates, and it is more convenient to employ the polar coordinate to handle the rotation and reflection operations.

In the remaining of this subsection, we employ the general procedure illustrated in Fig. 1 to obtain two group of local descriptors. The application to the translation operations gives us the local descriptors that have natural translational invariance within the same crystal pattern. In contrast, the local descriptors that are obtained by performing the rotation and reflection operations are not translationally invariant. We add a max-pooling step to

enforce the translational invariance, similar to the treatment adopted in the convolutional neural networks [13].

1.   Extracting features via translation operations

If the crystal patterns have different lattice vectors or orientations, we only need the translation symmetries to discriminate them. Based on the general procedure illustrated in Fig. 1, we calculate the translational-symmetry-based local descriptors as follows. We denote the translation shift as $T_{m,n}$, which shifts the patch by m pixels vertically and by n pixels horizontally. The transformed patch by this operation is simply the patch located at $(x + m, y + n)$, namely, $T_{m,n}P_{x,y} = P(x + m, y + n)$. Substituting this into Eq. (2), we obtain the symmetry score for the translation operation $T_{m,n}$,

$$
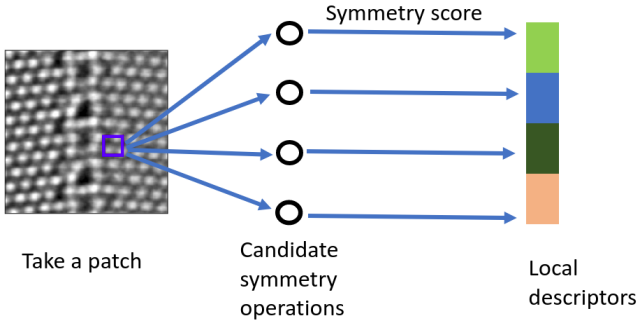\mathrm{SC}_{T_{m,n}}(x, y) = \mathrm{Corr}(P_{x,y}, P_{x+m,y+n}). \tag{3}
$$

FIG. 1: A schematic diagram to illustrate the symmetry-based local descriptors. There are four local-symmetry-feature extractors corresponding to four candidate symmetry operations in this diagram. We obtain the local descriptors of length four containing four symmetry features.
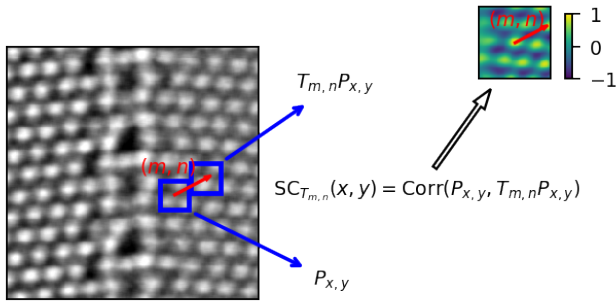

FIG. 2: Computing the local descriptors at pixel (x, y) based on the translation operations. The left plot: a real HAADF-STEM image of two copper grains. The right plot: the calculated local correlation map at pixel (x, y).

In practice, we confine the translation shift in a region of shape $(2w_x, 2w_y)$, i.e., $|m| \leq w_x$ and $|n| \leq w_y$, and select uniformly a fixed number N of translation shifts in this region. The symmetry scores for the selected translation shifts form a feature vector of length N, which is the local descriptor characterizing the local translational symmetries.

Before we move on to the discussion of the rotation-reflection-based local descriptors, it is worthwhile to first analyze what crystal patterns the translational-symmetry-based local descriptors are suitable to discriminate. Apparently, if the two crystal patterns have different Bravais lattices, lattice vectors, or orientations, we can employ the translational-symmety-based local descriptors to discriminate them easily. An interesting question is, can the translation-symmetry-based local descriptors discriminate easily the crystal patterns that have the same Bravais lattice, lattice vectors and orientation and different plane group symmetries? In order to answer this question, we take an extreme case in which we use all translation shifts in the region $|m| \leq w_x$ and

$|n| \leq w_y$ to form the local descriptors. In this case, the symmetry scores for all the translation shifts forms a two-dimensional map of shape $(2w_x + 1, 2w_y + 1)$, as shown in Fig. 2, which is denoted as the local correlation map in this paper. As the entry in the local-correlation map is evaluated through the correlation coefficient between the original and the translationally-shifted patch, the local-correlation map has the same symmetry with the auto-correlation function of the crystal pattern. The symmetry of the auto-correlation function of the crystals is denoted as the Patterson symmetry in the X-ray diffraction community, [14], and we follow this convention in this paper. The relationship between the seventeen plane-group symmetries and the seven Patterson symmetries is presented in Table I [14]. The local-correlation map can only have seven Patterson symmetries because of the extra constraints. First, there is always a two-fold rotational axis at the center. Second, the glide-reflection symmetry is not allowed, and is replaced by the corresponding reflection symmetry. Clearly, if the two crystal patterns with the same lattice vectors and orientation have the same Patterson symmetry whereas different plane-group symmetries, it becomes difficult for the translational-symmetry-based local descriptors to discriminate them. The other types of symmetries need to take into account for these microscopy images.

2. Extracting features via rotation and reflection operations

Following the procedure shown in Fig. 1, we present the implementation of the rotation-reflection-based local descriptors. In this case, it is more convenient to take the circular patches and employ the polar coordinates. The patches are rotated by a specific angle or reflected by a specific plane, and the symmetry score is again the Pearson's correlation coefficient between the original and the transformed patches, according to Eq. (2). We select a series of rotation angles $\{60°, 90°, 120°, 180°\}$, and also a series of reflection planes, and the symmetry score for each candidate symmetry operation represents a specific rotation or reflection symmetry feature. The local descriptors we obtain here cannot be used straightforwardly in the segmentation since they are not translationally invariant within the same crystal pattern. We present an example for a given reflection symmetry operation in Fig. 3. The symmetry scores show clearly different preferences to the crystal patterns on the left and right-hand sides. However, the symmetry scores are not homogeneously distributed within the crystal pattern indicating the point-group nature of the reflection and rotation symmetries. We add the max-pooling step to enforce the translational invariance, similar to the treatment in the convolutional neural networks [13]. As the max-pooling does not contain training parameters, our segmentation approach is still kept free of training.

TABLE I: The plane groups and the corresponding Patterson symmetries [14]

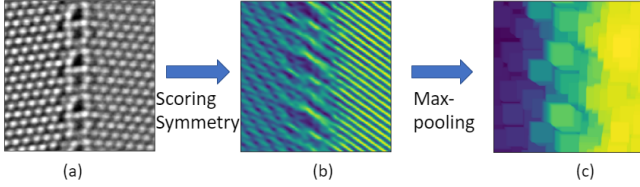| Plane group symmetry	p1	p2	p1m1	p1g1	p2mm	p2mg	p2gg	c1m1	c2mm
Corresponding Patterson symmetry	p2	p2	p2mm	p2mm	p2mm	p2mm	p2mm	c2mm	c2mm
Plane group symmetry	p4	p4mm	p4gm	p3	p6	p3m1	p31m	p6mm	
Corresponding Patterson symmetry	p4	p4mm	p4mm	p6	p6	p6mm	p6mm	p6mm	

FIG. 3: An example to illustrate the calculation of reflection-based local descriptors. The rotation-based local descriptors can be obtained in the similar way. (a) A real HAADF-STEM image of two copper grains. (b) The symmetry scores for a specific symmetry operation at all pixels. The color coding represents values of reflection symmetry scores. (c) The max-pooling enforces the translational invariance.

B. Principal component analysis

The local descriptor is a high-dimensional feature vector with each channel characterizing a specific symmetry. We may combine the information of all channels to get several features that capture the main differences of the crystal patterns in the image, and use these features as the input of the K-means clustering. With this treatment, we drop out the noisy information that is not important to discriminate the crystal patterns and reduce the computational cost in the latter clustering. In practice, we maximize an empirical metric, the variance, to obtain the optimal features. Intuitively, the larger the variance of the feature is, the more separate the crystal patterns are in the corresponding feature space. This metric works to obtain the optimal features in our case simply because the correlation between the separability and the variance of the features in our local descriptors, and a detailed discussion will be seen in section III. If we further assume that the optimal feature is a linear combination of all channels in the local descriptor, with the Euclidean norm of the combination coefficients bounded to be one, we immediately see that we can obtain them by simply projecting the local descriptors onto the principal components in PCA. We denote the optimal features obtained in this way as the PCA features, and employ the implementation of PCA in scikit-learn [15] in our code.

C. K-means clustering

We feed the PCA features into the k-means clustering algorithm [16, 17] to cluster the pixels in the feature space, which finds the cluster centers by minimizing the within-cluster variance and partition the feature space into Voronoi cells. Each cluster corresponds to a specific crystal pattern, and the pixels grouped into the same cluster are assigned with the same pattern label. We employ the implementation of the k-means clustering in scikit-learn [15] in our code, and the cluster centers are first initialized randomly and then optimized in an iterative way. The number of clusters is the hyperparameter that needs to be specified by the users.

D. Stride and upsampling

In many cases, it is not necessary to calculate the local descriptors of all pixels as they are highly correlated. We may first use the stride scheme to downsample the image, and then perform upsampling after the clustering is finished in order to match the size of the original image. In this way, we can reduce the computational cost drastically. The stride scheme we employed is the same with the one used in the convolutional neural networks. To be more specific, in the first step of our approach, instead of calculating the local descriptors for each pixel by shifting the local-symmetry-feature extractors every time by one unit, we shift them by a step size of s_x units vertically and by a step size of s_y units horizontally and only calculate the local descriptors for a fraction of the pixels. As the image is downsampled by the stride scheme, we add an upsampling scheme after the k-means clustering to expand the segmented image. We in the end obtain the segmented image that have the same size with the original one. In our upsampling scheme, we simply do an interpolation for the pixels that are not clustered based on the pattern labels of the clustered pixels, and the interpolated values are then rounded to integers.We note that both the stride scheme and the upsampling scheme contain no training parameters, which keeps our whole approach free of training.

III. APPLICATION AND DISCUSSION

In Fig. 4, we present an application to the HAADF-STEM image containing two phases (μ phase and Laves phase) of the iron-niobium intermetallics. Their crystal patterns in the specific projection plane have different two-dimensional Bravais lattices, one with the oblique lattice and the other with the rectangular lattice. The translation-symmetry-based local descriptors are ad-
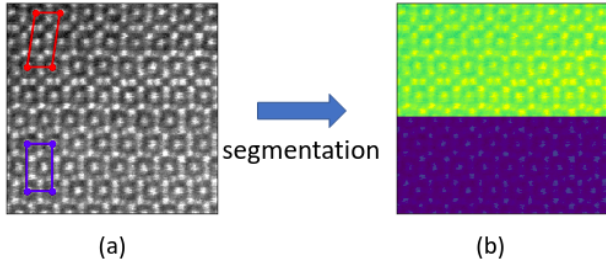
FIG. 4: (a) A HAADF-STEM image of two phases (μ phase and Laves phase) of the iron-niobium intermetallics that have different Bravais lattices. The unit cells of the two phases are shown in red and blue. (b) The image superimposed of the HAADF intensity and the pattern labels.
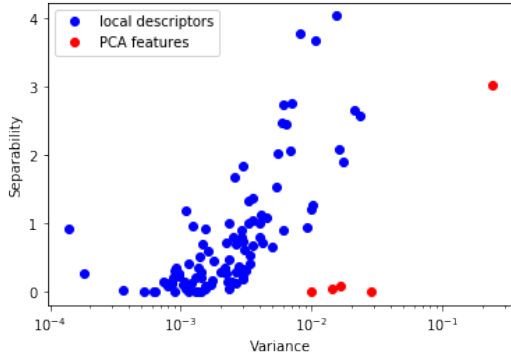

FIG. 5: The separability and variance of all features in the local descriptors and the five PCA features.
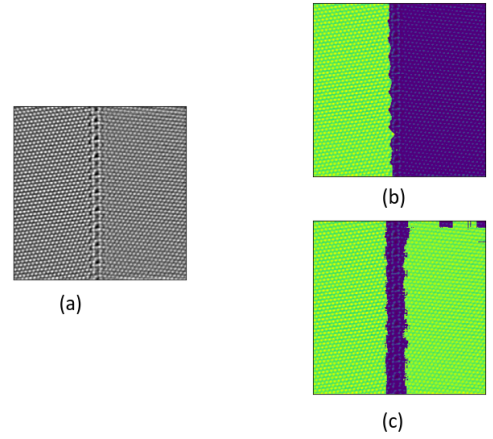

FIG. 6: (a) HAADF-STEM image containing two Cu single-crystal grains and a tilt grain boundary. (b) Segmentation via the reflection-based local descriptors. The image is superimposed of the HAADF intensity and the class labels. (c) Segmentation via the rotation-based local descriptors. The image is superimoposed of the HAADF intensity and the pattern labels.

strong correlation between the separability and the variance. The feature that has a larger variance also has a better separability. The PCA doesn't increase the separability, whereas it makes the energy more concentrated on the features that have better separability. We might think of the features with good separability the signal and those with bad separability the noise. The correlation between the separability and the variance is crucial for the unsupervised approach, which results in a good signal-noise ratio and the successful segmentation.

We show the second example in Fig. 6, in which we segment the HAADF-STEM image containing two differently oriented copper crystal grains and one tilt grain boundary with the reflection-symmetry-based and the rotation-symmetry-based local descriptors separately. The reflection-symmetry-based local descriptors partition this image into two segments, and the left and right crystal grains are clearly separated from each other. In contrast, the rotation-symmetry-based local descriptors group the two crystal grains into the same crystal pattern, and the crystal boundary is segmented into a different crystal pattern. The results indicate the influence of the choice of the local descriptors, as different local descriptors capture different symmetry information. In this example, the reflection-symmetry-based local descriptors are not rotationally invariant, and therefore, the same crystal grains with different orientations are partitioned into different crystal patterns. In contrast, the rotation-symmetry-based local descriptors are rotationally invariant and result in different interpretation of the crystal patterns in the image.

In our third example shown in Fig. 7, we segment the HAADF-STEM image for a nickel precipitate containing twin boundaries and stacking faults. We observed a very
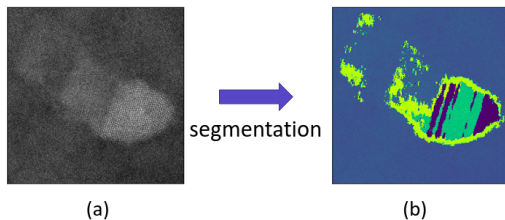
equate to discriminate them. We select uniformly one hundred translation operations in the region $|m| < 150$ pixels and $|n| < 150$ pixels to generate the local descriptors. The pattern labels are correctly assigned to the pixels in the upper and lower half of the image, as shown in Fig. 4.

We use this example to illustrate why our segmentation approach can work in an unsupervised manner. In order to show this, we calculate the Fisher separability [18] and variance for all features in the local descriptors and the five PCA features with largest variances. The Fisher separability is defined as the ratio between the between-class variance and the within-class variance [18], and can be evaluated according to

$$S = \frac{N_1(\mu_1 - \mu)^2 + N_2(\mu_2 - \mu)^2}{N_1\sigma_1^2 + N_2\sigma_2^2}, \quad (4)$$

where N_1 and N_2 are the number of samples in the first and the second class, μ_1 and μ_2 the within-class means, μ the total mean, and σ_1 and σ_2 the within-class variances. The separability tells how well the feature can discriminate the two crystal patterns, and the variance dictates the energy of the feature. We see that there is a

FIG. 7: (a) HAADF-STEM image for a nickel precipitate containing twin bounaries and stacking faults. (b) Image superimposed of the HAADF intensity and the pattern labels.

good spacial resolution, and the stacking faults and the twin boundaries are much better visible in the segmented image than in the original one.

We provide the jupyter notebooks for the calculations in this section [19], and all the figures might be reproduced with little effort.

IV. CONCLUSION

We present a semi-automated unsupervised machine-learning approach to segment the atomic-resolution microscopy images. In our approach, we extract the local descriptor via the symmetry operations, which forms an abundance feature vector to characterize the local symmetry information. We then use the principal component analysis (PCA) to reduce the dimension of the local descriptors, and observe in our example that the energy of the features with good separability is indeed increased by PCA. After that, we feed the PCA features into the k-means clustering algorithm in order to assign the pattern labels to the pixels. A stride and upsampling scheme is proposed in order to reduce the computational cost. We presented the successful application to the experimental HAADF-STEM images, and revealed one important feature of the unsupervised segmentation approach, the strong correlation between the feature's separability and energy (variance). We packaged our code as a python module, and released it in the github [20] and the python repository [19]. More tests can be found in the example folder in our github repository.

[1] P. Hansma, V. Elings, O. Marti, and C. Bracker, Science **242**, 209 (1988).

[2] A. Belianinov, R. Vasudevan, E. Strelcov, C. Steed, S. Yang, A. Tselev, S. Jesse, M. Biegalski, G. Shipman, C. Symons, et al., Advanced structural and chemical imaging **1** (2015).

[3] K. Kaufmann, C. Zhu, A. S. Rosengarten, D. Maryanovsky, T. J. Harrington, E. Marin, and K. S. Vecchio, Science **367**, 564 (2020).

[4] S. V. Kalinin, A. R. Lupini, O. Dyck, S. Jesse, M. Ziatdinov, and R. K. Vasudevan, MRS Bulletin **44**, 565 (2019).

[5] M. Ziatdinov, U. Fuchs, J. H. G. Owen, J. N. Randall, and S. V. Kalinin, *Robust multi-scale multi-feature deep learning for atomic and defect identification in scanning tunneling microscopy on h-si(100) 2x1 surface* (2020).

[6] T. Sadowski, C. C. Broadbridge, and J. Daponte, MRS Proceedings **982** (2006).

[7] M. Mancas, B. Gosselin, and B. Macq, in *Image Processing: Algorithms and Systems IV*, edited by E. R. Dougherty, J. T. Astola, and K. O. Egiazarian, International Society for Optics and Photonics (SPIE, 2005), vol. 5672, pp. 388 – 398.

[8] S. Navlakha, P. Ahammad, and E. W. Myers, BMC Bioinformatics **14** (2013).

[9] R. Nock and F. Nielsen, IEEE Transactions on Pattern Analysis and Machine Intelligence **26**, 1452 (2004).

[10] I. Arganda-Carreras, V. Kaynig, C. Rueden, K. W. Eliceiri, J. Schindelin, A. Cardona, and H. Sebastian Seung, Bioinformatics **33**, 2424 (2017).

[11] Q. D. Vu, S. Graham, T. Kurc, M. N. N. To, M. Shaban, T. Qaiser, N. A. Koohbanani, S. A. Khurram, J. Kalpathy-Cramer, T. Zhao, et al., Frontiers in Bioengineering and Biotechnology **7**, 53 (2019).

[12] N. Borodinov, W.-Y. Tsai, V. V. Korolkov, N. Balke, S. V. Kalinin, and O. S. Ovchinnikova, Applied Physics Letters **116**, 044103 (2020).

[13] Y. Boureau, J. Ponce, and Y. Lecun, in *ICML 2010 - Proceedings, 27th International Conference on Machine Learning* (2010), ICML 2010 - Proceedings, 27th International Conference on Machine Learning, pp. 111–118, ISBN 9781605589077.

[14] IUCr, *International Tables for Crystallography, Volume A: Space Group Symmetry*, International Tables for Crystallography (Kluwer Academic Publishers, 2002), 5th ed.

[15] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al., Journal of Machine Learning Research **12**, 2825 (2011).

[16] J. MacQueen, in *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics* (University of California Press, Berkeley, Calif., 1967), pp. 281–297, URL https://projecteuclid.org/euclid.bsmsp/1200512992.

[17] D. Arthur and S. Vassilvitskii, in *In Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms* (2007).

[18] R. A. Fisher, Annals of Eugenics **7**, 179 (1936).

[19] *https://pypi.org/project/pystem/*.

[20] *https://github.com/ningwang1990/pystem*.