



Cite this: *Mater. Adv.*, 2025, 6, 6820

## GRATEv2: computational tools for real-time analysis of high-throughput high-resolution TEM (HRTEM) images of conjugated polymers

Dhruv Gamdha,<sup>a</sup> Ryan Fair,<sup>c</sup> Adarsh Krishnamurthy,<sup>ab</sup> Enrique D. Gomez  <sup>cd</sup> and Baskar Ganapathysubramanian  <sup>ab</sup>

Automated analysis of high-resolution transmission electron microscopy (HRTEM) images is increasingly essential for advancing research in organic electronics, where precise characterization of lamellar, one-dimensional crystalline domains in conjugated polymers governs device performance. This paper introduces an open-source computational framework—GRATEv2 (GRaph-based Analysis of TEM, Version 2)—designed for near-real-time analysis of semi-crystalline, polymeric microstructures; its capabilities are illustrated on poly[N-9'-heptadecanyl-2,7-carbazole-*alt*-5,5-(4',7'-di-2-thienyl-2',1',3'-benzothiadiazole)] (PCDTBT), a benchmark material in organic photovoltaics. GRATEv2 employs fast, automated image processing algorithms, enabling rapid extraction of structural features like *d*-spacing, orientation, and crystal shape metrics. Bayesian optimization rapidly identifies the parameters (that are traditionally user-defined) in the approach, reducing the need for manual parameter tuning and thus enhancing reproducibility and usability. Additionally, GRATEv2 is compatible with high-performance computing (HPC) environments, allowing for efficient, large-scale data processing at near real-time speeds. A unique feature of GRATEv2 is a Wasserstein distance-based stopping criterion, which optimizes data collection by determining when further sampling no longer adds statistically significant information. This capability optimizes the amount of time the TEM facility is used while ensuring data adequacy for in-depth analysis. Open-source and tested on a substantial PCDTBT dataset, this tool offers a powerful, robust, and accessible solution for high-throughput material characterization in organic electronics.

Received 28th April 2025,  
Accepted 12th August 2025

DOI: 10.1039/d5ma00409h

[rsc.li/materials-advances](http://rsc.li/materials-advances)

## 1 Introduction

Microscopy has long been a cornerstone in materials science, offering a unique window into the microstructure and enabling scientists to study properties at various scales, from the millimeter down to the atomic level.<sup>1,2</sup> By visualizing otherwise inaccessible structures, microscopy provides critical insights into how atomic and molecular arrangements influence key macroscopic properties such as mechanical strength, electrical conductivity, and chemical reactivity.<sup>3</sup> This fundamental understanding is essential for designing advanced materials with tailored properties for applications in fields such as electronics, energy storage, and catalysis.

High-resolution transmission electron microscopy (HRTEM) has advanced significantly in recent years, transforming

nanoscale imaging and allowing researchers to capture atomic-level details of materials.<sup>4</sup> HRTEM can now achieve sub-angstrom spatial resolution, making it possible to directly observe atomic lattices, defects, and interfaces that govern a material's behavior.<sup>5</sup> The development of automated data acquisition systems has further broadened HRTEM's capabilities, enabling the collection of extensive datasets comprising hundreds or even thousands of high-resolution images.<sup>6</sup> These technological advances have opened new avenues for studying complex materials, such as organic semiconductors and conjugated polymers, with applications in organic electronics and photovoltaics.<sup>7</sup>

While modern HRTEM has enhanced our ability to study complex materials, it also introduces challenges in data management and analysis due to the sheer volume and complexity of the data generated.<sup>8</sup> The scale of high-resolution datasets can quickly overwhelm traditional analysis workflows, which are often manual and time-consuming. A particularly acute bottleneck is the creation of annotation masks that serve as "ground truth" for algorithm development: drawing them is labor-intensive, and the outcome can vary from one expert to

<sup>a</sup> Department of Mechanical Engineering, Iowa State University, Ames, IA, USA.  
E-mail: [baskarg@iastate.edu](mailto:baskarg@iastate.edu)

<sup>b</sup> Translational AI Research Center (TrAC), Iowa State University, Ames, IA, USA

<sup>c</sup> Department of Chemical Engineering, The Pennsylvania State University, University Park, PA, USA

<sup>d</sup> Department of Material Science and Engineering, The Pennsylvania State University, University Park, PA, USA



another. This manual approach is highly dependent on the expertise and subjective judgment of the experimentalist, making it challenging to ensure consistency and reproducibility. In high-throughput applications where rapid feedback is essential, such as optimizing synthesis conditions or tracking real-time structural changes, the limitations of manual analysis become particularly pronounced.

To address these challenges, automated methods for HRTEM data analysis have emerged over the past decade. These methods aim to extract quantitative structural information from digital micrographs with minimal human intervention, improving both the efficiency and reliability of analysis.<sup>3,9–12</sup> A motivating example is our prior work, GRATE,<sup>13</sup> which served as a proof-of-concept for using graph-based algorithms to identify crystalline regions by first thinning fringes into skeletons and then clustering them based on proximity and orientation.<sup>13</sup> However, this first version was limited by its implementation in MATLAB, restricting it to offline post-processing, and it relied on sensitive manual tuning of its parameters, which hindered reproducibility.<sup>13</sup> GRATEv2 significantly advances this foundation through a complete redesign in Python for HPC compatibility and several key innovations. The image processing pipeline is refined with additional preprocessing steps and a more robust thresholding approach, and introduces an explicit step of segmenting skeletons into uniform “bones” to regularize the graph construction. Most critically, GRATEv2 replaces the manual tuning process with automated Bayesian Optimization and adds a novel Wasserstein distance-based stopping criterion to guide data collection, transforming the original concept into a reproducible, high-throughput framework.

In recent years, there has been a strong push toward developing *in situ*, or real-time, automated analysis methods.<sup>14–16</sup> *In situ* analysis allows for data interpretation during experiments, providing immediate feedback that can guide adjustments in experimental parameters. This capability is particularly valuable in dynamic experiments, such as observing structural changes in response to external stimuli or monitoring materials during synthesis.<sup>17–23</sup> Real-time analysis requires methods that can handle high-resolution data quickly and accurately, maintaining performance under the demanding conditions of live data acquisition.

Various automated analysis philosophies have been explored. Machine learning tools like the Trainable Weka Segmentation plugin for Fiji/ImageJ<sup>24</sup> offer a powerful, user-friendly approach for general pixel classification by training a classifier on a small number of user-provided labels.<sup>25</sup> However, such tools typically rely on a generic set of image filters and are designed for classifying regions or simple objects like nanoparticles rather than identifying and connecting the long, thin, and often discontinuous fringe patterns that constitute a single crystalline domain in polymers. For such specialized tasks, bespoke rule-based image processing pipelines are often more effective. A key challenge, however, is the tuning of their many interacting parameters. Recently, Bayesian Optimization has been powerfully applied to automate this tuning process. In a notable example, Barakati *et al.*<sup>26</sup> engineered a “physics-based reward function,” which uses physical priors (*e.g.*, expected atom

density) to serve as the objective for the optimization, thereby avoiding the need for manual annotations.<sup>26</sup>

GRATEv2 shares the philosophy of Barakati *et al.*<sup>26</sup> of using Bayesian Optimization to tune a classical pipeline but is novel in two critical areas: the algorithm being optimized and the nature of the objective function. First, while the work by Barakati *et al.*<sup>26</sup> focuses on optimizing standard algorithms like Laplacian-of-Gaussian for atom-finding, GRATEv2 optimizes a novel, graph-based pipeline specifically designed to segment the unique lamellar structures of polymer films. Second, we propose a different approach for the optimization’s objective. Barakati *et al.*<sup>26</sup> engineer a clever “physics-based reward” that avoids manual labeling by using physical priors. For the complex and irregular morphology of polymer crystallites, defining such simple physical rules is challenging. Instead, GRATEv2 uses a small set of manually annotated images to compute the Intersection-over-Union (IoU), providing a direct and robust way to ensure the final segmentation matches expert perception.

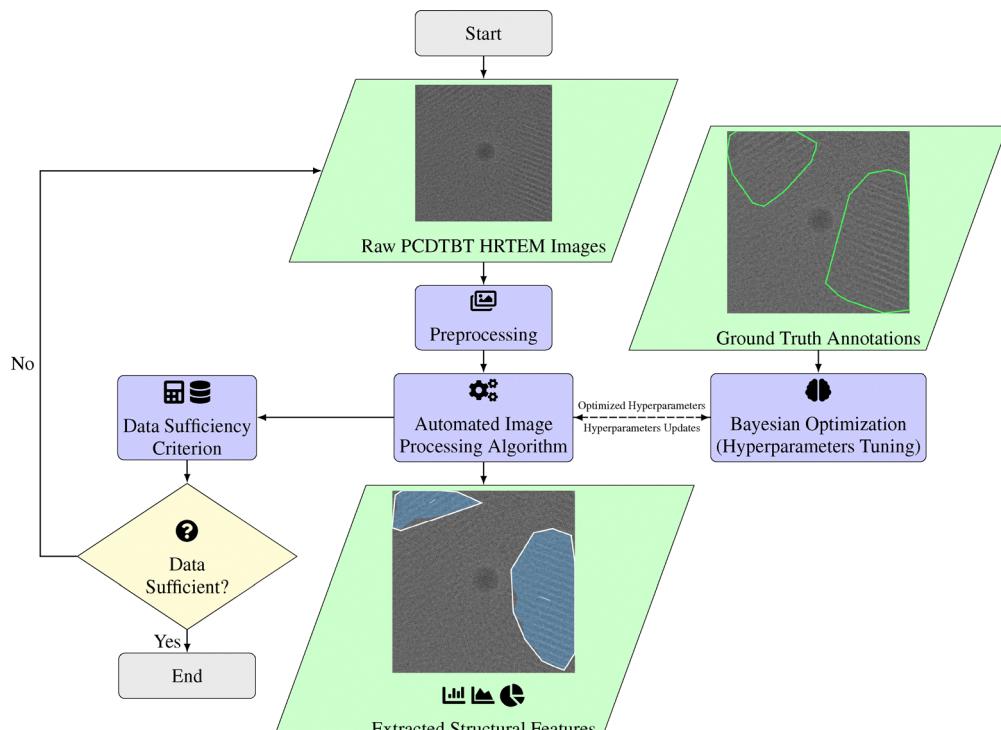
GRATEv2 aims to bridge the gap between offline and real-time analysis in high-resolution transmission electron microscopy (HRTEM) by providing an automated, image-processing-based framework with minimal human intervention. Tailored for high-throughput settings, GRATEv2 combines rapid data extraction with a robust and user-friendly parameter optimization process. By focusing on image processing techniques augmented with Gaussian process optimization, GRATEv2 minimizes the need for manual parameter selection and tuning, enhancing reproducibility and accessibility for researchers.

A schematic overview of the GRATEv2 computational framework is shown in Fig. 1a. The framework processes raw HRTEM images of PCDTBT (Fig. 2a), applies preprocessing, and performs automated image processing with parameters optimized *via* Bayesian optimization (Fig. 1b). Traditionally, manual annotation of crystals in HRTEM images is time-consuming and subjective (Fig. 2). In our approach, only a dozen manually annotated images (Fig. 2b) are used as input to a Bayesian optimizer to rapidly identify material-specific image processing parameters. This minimal requirement significantly reduces the burden on researchers, facilitating rapid deployment of the algorithm on new datasets.

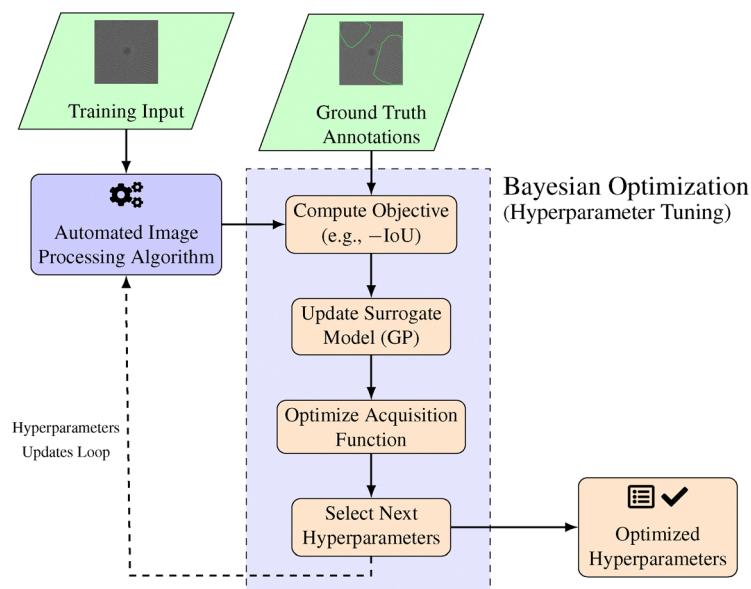
GRATEv2 introduces several key innovations to address the limitations of current HRTEM analysis methods:

1. GRATEv2 offers rapid processing of HRTEM micrographs that exhibit lamellar, 1-D fringe patterns typical of semi-crystalline conjugated polymers, delivering results in a few seconds per image and supporting batch multiprocessing for large datasets.
2. It brings a graph-based, image-processing pipeline to organic polymer systems such as PCDTBT, where atomic-resolution lattice models are neither required nor assumed.
3. Bayesian optimization is employed within GRATEv2 to automate the tuning of material-specific image processing parameters, allowing the framework to adapt to different datasets with minimal expert input (Fig. 1b).
4. The algorithm parameters are constructed as functions of known *d*-spacing values, simplifying parameter selection and making the method more accessible to users without extensive





(a) Schematic overview of the GRATEv2 computational framework.



(b) Detailed diagram of the Bayesian Optimization process.

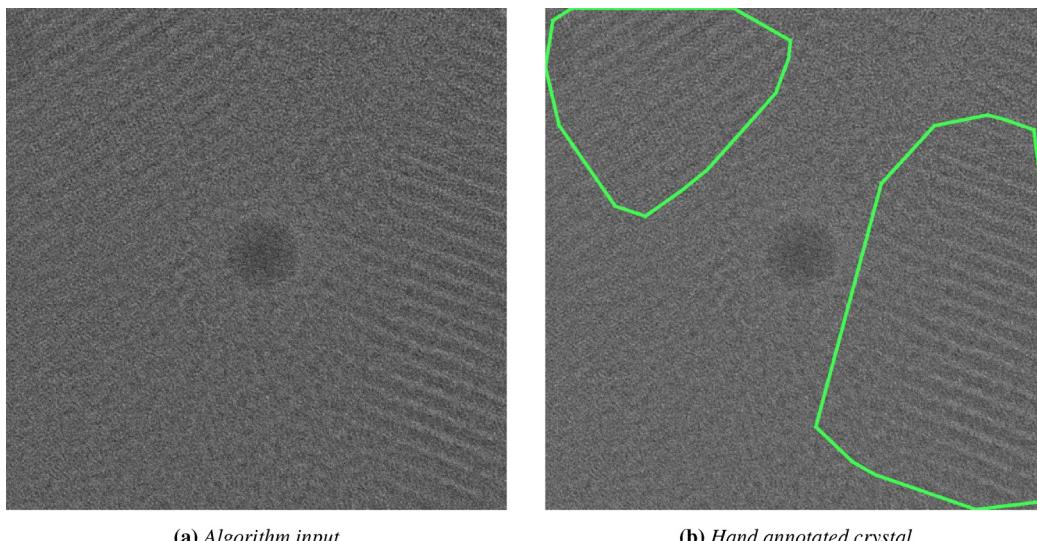
**Fig. 1** Schematic overview of the GRATEv2 computational framework. The framework processes raw HRTEM images of PCDTBT, applies preprocessing, and performs automated image processing with parameters optimized via Bayesian optimization. A data sufficiency criterion based on the Wasserstein distance assesses whether additional TEM data is needed. The output comprises extracted structural features such as  $d$ -spacing, orientation, and crystal shape metrics. (a) The overall computational framework of GRATEv2, and (b) the detailed Bayesian Optimization (BO) component used for parameter tuning.

image processing expertise. This also ensures that parameter selection is interpretable and, thus, scientifically justified.

5. GRATEv2 incorporates a data sufficiency criterion based on the Wasserstein distance to guide data collection efforts.

This criterion provides a quantitative stopping point, indicating when further TEM data collection no longer yields additional insights, which is important when access to imaging facilities is limited and imaging is expensive.





**Fig. 2** Example input to the optimisation workflow. (a) Raw HRTEM micrograph supplied to the algorithm. (b) The same micrograph overlaid with hand-drawn crystal contours that serve as ground-truth masks for evaluating candidate parameter sets during Bayesian optimisation.

By optimizing data collection, GRATEv2 helps experimentalists avoid unnecessary resource expenditure while ensuring data quality. This combination of fast, automated processing, real-time adaptability, and efficient data collection positions GRATEv2 as a powerful tool for materials research, particularly in the study of organic electronic materials such as conjugated polymers.<sup>27–29</sup>

## 2 GRATEv2 framework and algorithms

### 2.1 HRTEM sample preparation and measurement method

PCDTBT was synthesized using previously published procedures for Suzuki polycondensation in a Schlenk reactor flask.<sup>30,31</sup> Sigma-Aldrich supplied all reactants. Polymerization occurred between 9-(9-heptadecanyl)-9H-carbazole-2,7-diboronic acid bis(pinacol) ester and 4,7-bis(2-bromo-5-thienyl)-2,1,3-benzothiadiazole in toluene, with equimolar amounts of each monomer. All other synthesis and purification procedures remained unaltered compared to the cited sources. The synthesis product was characterized by  $H^1$  nuclear magnetic resonance at 500 MHz.<sup>32</sup>

5 mg mL<sup>-1</sup> solutions of PCDTBT and chlorobenzene (Sigma-Aldrich) were prepared in a nitrogen glovebox and mixed overnight on a hotplate at 45 °C. Silicon wafers were sonicated in acetone for 20 minutes, then isopropanol for 20 minutes. The wafers then underwent 20 minutes of ultraviolet light ozonation. A PEDOT:PSS (Clevios P and H. C. Starck) and water solution were then spin-coated onto the clean substrates at 4000 RPM for 2 minutes. The samples were then brought into the glovebox, and the heated PCDTBT solution was spin-coated on top of the PEDOT:PSS layer at 800 RPM for 2 minutes. The sample was then cut into squares and floated off in the water, and samples were collected on copper TEM grids. Samples were left to dry overnight and then annealed inside a nitrogen glovebox.

High-resolution imaging experiments were conducted on the Titan Krios microscope at the Penn State Materials

Characterization Laboratory. The accelerating voltage was 300 kV, and the detector was a Falcon 3EC direct electron detector in counted mode. Regions of interest were spaced 2.5  $\mu$ m apart and visually inspected on the atlas image for tears and defects before acquisition. The spot size was set to 5, and autofocus was done at 300k $\times$  magnification before being increased to 470k $\times$  for acquiring a 2.5 second exposure. The microscope produced a 650 nm beam with a dose rate of 50 e  $\text{\AA}^2$  s.

### 2.2 Quantities of interest

The inputs to the algorithm are (1) the HRTEM image (2) approx  $d$ -spacings of the crystals to detect (3) the resolution of the image (4) process parameters. The algorithm is designed to detect crystals in the HRTEM images and extract their features. The algorithm outputs the segmentation result for each input and a CSV file containing the feature details such as  $d$ -spacing, orientation, and crystal shape metrics. The algorithm is capable of detecting multiple crystals in a single image, and the output is a list of detected crystals with their corresponding features.

We report two notions of separation as described in Table 1 and eqn (1) and (2):

$$D_{\text{metric}} = \frac{d_{\text{center-center}}}{r_1 + r_2} \quad (1)$$

$$D_{\text{direct}} = d_{\text{center-center}} \quad (2)$$

Here  $d_{\text{center-center}}$  is the Euclidean distance between the centroids of two crystals. Because actual crystal outlines are irregular, each area  $A_i$  (obtained from the  $\alpha$ -shape) is first converted to an equivalent circular radius  $r_i = \sqrt{A_i/\pi}$ .

### 2.3 Image processing algorithm

Fig. 3 illustrates the flowchart of GRATEv2 algorithm along with the associated process parameters. The process parameters are



**Table 1** Description of various distances and radii used in the algorithm. The metric distance  $D_{\text{metric}}$  is a dimensionless, size-normalised measure of separation between crystals, while the direct distance  $D_{\text{direct}}$  is the raw centroid–centroid spacing. The equivalent radii  $r_1$  and  $r_2$  are derived from the  $\alpha$ -shape areas of the crystals. These definitions are crucial for understanding the spatial relationships between crystalline domains in HRTEM images

Symbol	Description
$D_{\text{metric}}$	Metric distance—a dimensionless, size-normalised measure of separation. • $D_{\text{metric}} \approx 1$ : distance comparable to the combined crystal sizes. • $D_{\text{metric}} > 1$ : crystals farther apart than their sizes suggest (free space between them). • $D_{\text{metric}} < 1$ : crystals closer together (or partly overlapping) relative to their sizes.
$D_{\text{direct}}$	Direct distance (nm)—the raw centroid–centroid spacing.
$d_{\text{center-center}}$	Straight-line centroid distance (nm).
$r_1, r_2$	Equivalent radii (nm)— $\sqrt{A_i/\pi}$ with $A_i$ the $\alpha$ -shape area of crystal $i$ .

indicated in blue boxes, while the intermediate steps corresponding to Fig. 4 are marked with brown numeric annotations.

The algorithm parameters are classified into two categories: primary parameters and secondary parameters. The primary parameters are independent values provided by the user based on the dataset's imaging conditions and material properties, specifically the crystal  $d$ -spacing value and the image resolution. These parameters scale the key process parameters, enabling the algorithm to generalize across different imaging conditions and crystals of interest. They may vary significantly between datasets. Secondary parameters correspond to individual steps of the algorithm and are set to optimal values; they may require fine-tuning when changing datasets.

The algorithm is designed with several key objectives in mind: to generate a clear and concise representation of the essential information in the image; to enhance data quality by filtering out noise and irrelevant details; to convert the processed image data into a graph structure for advanced analysis; to use graph algorithms to identify clusters of polymer backbones forming crystallites; and to apply Fourier transform techniques to determine the  $d$ -spacing values.

Each step of the algorithm is detailed below, along with its associated parameters.

1. Initially, blurring is applied to smooth the image and reduce sharpness, which helps in minimizing noise and preparing the image for subsequent processing. This operation is performed multiple times, with the number of blurring iterations specified as a parameter. Following blurring, histogram equalization is employed to increase the image contrast, making it sharper and enhancing the distinction between polymer chains and the background. Improved contrast facilitates more accurate thresholding in the subsequent step.

2. The next step involves image thresholding using Otsu's method to create a binary representation of the image, where polymer chains appear as black regions and the background as white. The result of this step is shown in Fig. 4b. After thresholding, morphological closing and opening operations are performed to remove small black and white spots considered as noise. The kernel sizes for closing and opening are parameters that do not depend on the  $d$ -spacing and are set based on the noise characteristics of the image.

3. Subsequently, we perform skeletonization and branching. Skeletonization reduces the polymer regions to single-pixel-wide lines, representing the skeleton of the polymer chains.

Branching breaks the skeleton at junctions where three or more connections occur, resulting in branched skeletonized representations called backbones. The output of this step is depicted in Fig. 4c.

4. Following skeletonisation, we remove spurious fragments by discarding any backbone whose pixel length is less than a length threshold proportional to the lattice  $d$ -spacing (in pixels). The proportionality constant is supplied by the user as a tunable parameter. This scale-aware filter suppresses noise while retaining only those backbones long enough to represent meaningful crystal fringes.

5. Each surviving backbone is traversed pixel-by-pixel from one end; after every segment length set proportional to the  $d$ -spacing in pixels—where the proportionality constant is a user-specified parameter—the backbone is severed by flipping that pixel from black to white. Because skeletons are one pixel wide, this procedure divides the polyline into uniform-length bones. Linking the segment length to  $d$ -spacing provides elasticity over the expected range of lattice spacings. The effect of filtering and segmentation is illustrated in Fig. 4d.

6. For each bone, we perform ellipse construction by fitting an ellipse to its pixel locations using the scikitimage library.<sup>33</sup> This provides the major and minor axes of the bone, which are used for further analysis. Ellipses are preferred because they help filter out non-linear bones (curved structures not part of crystalline regions) and facilitate the creation of a graph representation where each ellipse serves as a node.

7. To identify the crystalline regions, we apply ellipse aspect ratio filtering. Crystalline regions are composed of linear polymer backbones, so we filter out curved bones by setting a threshold on the ellipse aspect ratio (major axis length divided by minor axis length). Bones with aspect ratios above this threshold are retained for further analysis. This step's output is shown in Fig. 4e.

8. A graph is then constructed where each ellipse represents a node. An edge is created between two nodes if the distance between their centers is less than the adjacency distance parameter (proportional to the  $d$ -spacing) and the angle between their major axes is less than the adjacency angle parameter (also proportional to the  $d$ -spacing). The graph is stored as an adjacency matrix.

9. Connected-component clustering. A depth-first search (DFS) identifies all connected components (CCs). CCs with fewer than a threshold number of nodes are treated as noise



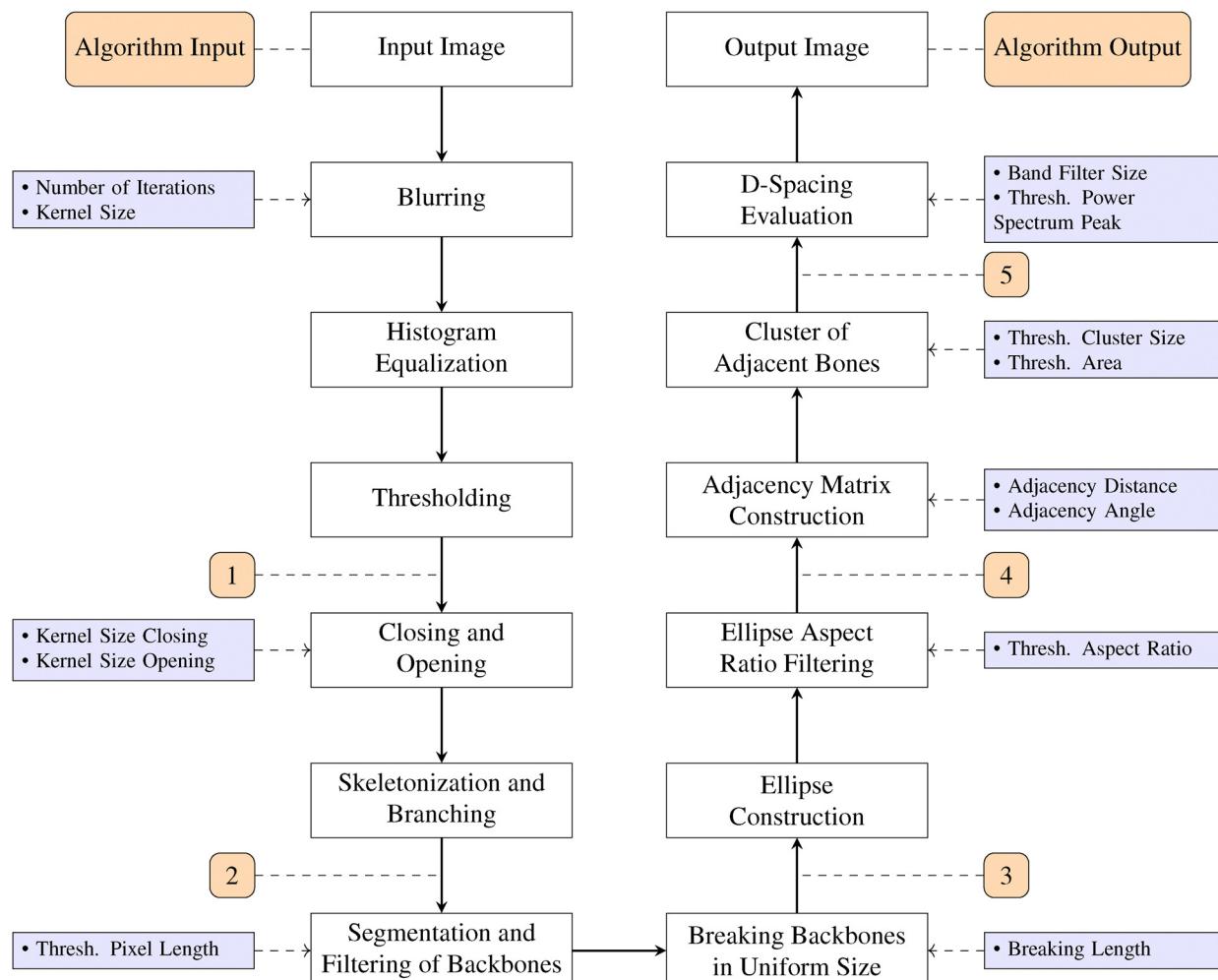


Fig. 3 Algorithm flowchart with parameters. Blue boxes are the process parameters and brown boxes are the intermediate steps shown in Fig. 4.

and discarded; the rest are considered candidate crystals (Fig. 4f).

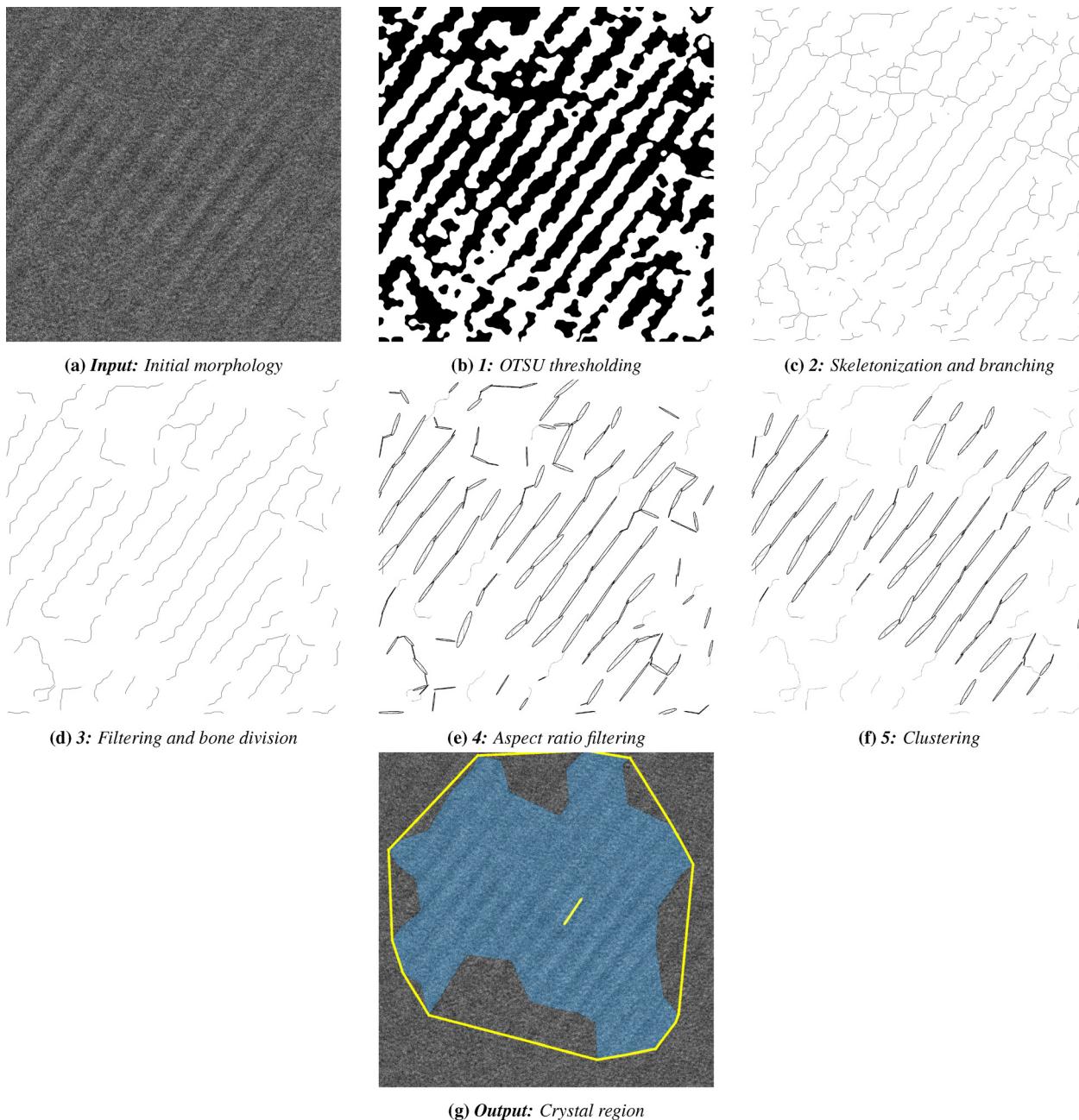
10. Boundary extraction and overlay. For each retained CC we collect the end-points of the major axes of its ellipses, yielding a point cloud that captures the crystals footprint. We compute both a convex hull (`scipy.spatial.ConvexHull`) and an  $\alpha$ -shape (`alphashape`) from this cloud, then overlay them on the original micrograph to visualise the detection (Fig. 4g).

11. Per-crystal metric extraction. From each crystal boundary we record area, centroid, major/minor axis lengths and orientation. A line indicating the principal lattice direction is also drawn on the micrograph; these metrics feed the statistical analysis described in Section 3.

Finally, we perform  $d$ -spacing evaluation for each detected crystal region. The largest possible square region within the crystal is selected, and a Fast Fourier Transform (FFT) is performed to transform the spatial domain into the frequency domain. Band-pass filtering is applied to remove frequencies outside the range of interest. The location of the peak frequency above a set threshold is used to calculate the exact  $d$ -spacing value and the orientation of the crystal pattern. The frequency

threshold is a parameter. The evaluation of  $d$ -spacing using FFT is illustrated in Fig. 5. The algorithm outputs include visualizations such as convex hulls and alpha shapes of detected crystal regions overlaid on the original image, as well as quantitative data like area, centroid, major and minor axis lengths, orientation, and  $d$ -spacing values for each crystal. All features are saved in CSV files for further analysis. These outputs provide valuable insights into the material's microstructure and can aid in understanding material properties more effectively.

The algorithm is implemented in Python, utilizing the scikit-image library for image processing and SciPy for computational functions. A configuration file is used to input all relevant parameters, dataset paths, and result paths. The code is tested on Ubuntu Linux-based local systems and HPC servers. It supports batch processing and multiprocessing, allowing for efficient analysis of large datasets. Results are stored in a well-organized directory structure, with options to save intermediate outputs for debugging purposes. Comprehensive documentation is provided, detailing code usage and parameter settings. A summary of the primary and secondary parameters used in the algorithm is provided in Appendix A. These parameters are



**Fig. 4** Intermediate and final outputs of the pipeline (see the flowchart in Fig. 3). (a) Input: initial morphology. (b) Step 1—Otsu thresholding (corresponds to orange callout 1 in Fig. 3). (c) Step 2—skeletonization and branching (callout 2). (d) Step 3—filtering short backbones and division into uniform bones (callout 3). (e) Step 4—aspect-ratio filtering of ellipses (callout 4). (f) Step 5—clustering of adjacent bones (callout 5). (g) Output: detected crystal region with boundary/principal-direction overlay.

optimized for detecting crystals in the HRTEM dataset of PCDTBT organic photovoltaic materials.

#### 2.4 Bayesian optimization

Optimizing hyperparameters in image processing algorithms is crucial for enhancing performance, especially in complex tasks like automated detection of crystalline domains in HRTEM images. Traditional methods of parameter tuning, such as grid search or manual adjustment, can be inefficient and may not guarantee optimal results due to the high dimensionality and

computational expense. To address this challenge, we integrated Bayesian optimization into our framework, GRATEv2, to systematically and efficiently identify the optimal set of parameters.

Bayesian optimization is a sequential design strategy for global optimization of black-box functions that are expensive to evaluate especially due to large hyperparameter space.<sup>34</sup> It builds a probabilistic model of the objective function and uses it to select the most promising hyperparameters to evaluate next, balancing exploration and exploitation.



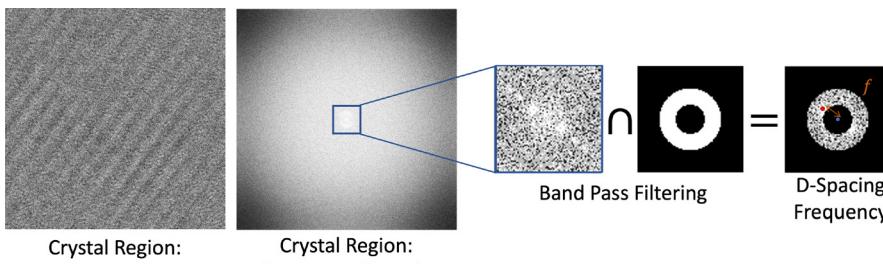


Fig. 5 Evaluation of  $d$ -spacing for the crystal using the fast Fourier transform.

**2.4.1 Mathematical formulation.** Let  $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^d$  denote a vector of  $d$  hyperparameters, and let  $f: \mathcal{X} \rightarrow \mathbb{R}$  be the objective function that maps hyperparameters to a scalar performance metric we wish to minimise. In GRATEv2 the objective is the negative mean Intersection-over-Union (IoU) between predicted and ground-truth crystal masks; equivalently we maximise IoU:

$$\mathbf{x}^* = \arg \min_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}), \quad f(\mathbf{x}) = -\frac{1}{N} \sum_{i=1}^N \text{IoU}(\mathcal{P}_i(\mathbf{x}), \mathcal{G}_i). \quad (3)$$

For a single image the IoU (also known as the Jaccard index) is defined as:

$$\text{IoU}(\mathcal{P}, \mathcal{G}) = \frac{|\mathcal{P} \cap \mathcal{G}|}{|\mathcal{P} \cup \mathcal{G}|}, \quad (4)$$

where  $\mathcal{P}$  is the set of pixels predicted as crystalline and  $\mathcal{G}$  is the corresponding ground-truth set. Fig. 6 gives a visual illustration. An IoU of 1.0 means perfect overlap; 0.0 denotes no overlap.

**2.4.2 Gaussian process surrogate model.** Bayesian optimization relies on a surrogate model to approximate the objective function. We use a Gaussian process (GP) prior<sup>35</sup> over functions to model  $f(\mathbf{x})$ . A GP is defined by its mean function  $m(\mathbf{x})$  and covariance function  $k(\mathbf{x}, \mathbf{x}')$ :

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')). \quad (5)$$

We assume a zero mean function  $m(\mathbf{x}) = 0$  without loss of generality, and select a suitable covariance function, such as the squared exponential (radial basis function) kernel:

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left(-\frac{1}{2}(\mathbf{x}, \mathbf{x}')^\top \mathbf{L}^{-1}(\mathbf{x}, \mathbf{x}')\right), \quad (6)$$

where  $\sigma_f^2$  is the signal variance and  $\mathbf{L}$  is a diagonal matrix of

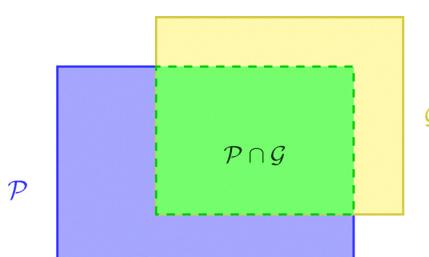


Fig. 6 Illustration of Intersection-over-Union (IoU). Blue represents the predicted crystalline mask  $\mathcal{P}$ , yellow the ground-truth mask  $\mathcal{G}$ , and green their intersection  $\mathcal{P} \cap \mathcal{G}$ . The IoU is the ratio of the green area to the total area covered by either mask  $|\mathcal{P} \cup \mathcal{G}|$ .

length-scale parameters  $\ell_i^2$ , controlling the smoothness of the function along each dimension.

**2.4.3 Posterior distribution.** Given a set of  $n$  observed data points  $\mathcal{D}_n = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ , where  $y_i = f(\mathbf{x}_i) + \epsilon_i$  and  $\epsilon_i \sim \mathcal{N}(0, \sigma_n^2)$  is Gaussian observation noise, the GP posterior predictive distribution at a new point  $\mathbf{x}_*$  is given by:

$$\mu_n(\mathbf{x}_*) = \mathbf{k}_*^\top (\mathbf{K} + \sigma_n^2 \mathbf{I})^{-1} \mathbf{y}, \quad (7)$$

$$\sigma_n^2(\mathbf{x}_*) = k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_*^\top (\mathbf{K} + \sigma_n^2 \mathbf{I})^{-1} \mathbf{k}_*, \quad (8)$$

where  $\mathbf{k}_* = [k(\mathbf{x}_1, \mathbf{x}_*), \dots, k(\mathbf{x}_n, \mathbf{x}_*)]^\top$ ,  $\mathbf{K}$  is the  $n \times n$  covariance matrix with  $[\mathbf{K}]_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$ , and  $\mathbf{y} = [y_1, \dots, y_n]^\top$ .

**2.4.4 Acquisition function.** An acquisition function  $\alpha(\mathbf{x}; \mathcal{D}_n)$  guides the selection of the next evaluation point by quantifying the utility of sampling at  $\mathbf{x}$ . We use the expected improvement (EI) acquisition function,<sup>36</sup> defined as:

$$\alpha_{\text{EI}}(\mathbf{x}) = \mathbb{E}[\max(0, f_{\min} - f(\mathbf{x}))], \quad (9)$$

where  $f_{\min}$  is the minimum observed value of the objective function. Under the GP posterior, the EI can be computed analytically:

$$\alpha_{\text{EI}}(\mathbf{x}) = (f_{\min} - \mu_n(\mathbf{x})) \Phi\left(\frac{f_{\min} - \mu_n(\mathbf{x})}{\sigma_n(\mathbf{x})}\right) + \sigma_n(\mathbf{x}) \phi\left(\frac{f_{\min} - \mu_n(\mathbf{x})}{\sigma_n(\mathbf{x})}\right), \quad (10)$$

where  $\mu_n(\mathbf{x})$  and  $\sigma_n(\mathbf{x})$  are the posterior mean and standard deviation from eqn (7) and (8),  $\Phi(\cdot)$  is the standard normal cumulative distribution function, and  $\phi(\cdot)$  is the standard normal probability density function.

**2.4.5 Optimization loop.** As shown in Fig. 7, the Bayesian optimization algorithm proceeds iteratively through initialization, surrogate model updates, and acquisition function optimization.

The Bayesian optimization algorithm begins with the initialization step, where the objective function is evaluated at an initial set of hyperparameters  $\{\mathbf{x}_i\}_{i=1}^{n_0}$ , often chosen via a space-filling design like Latin hypercube sampling. Next, a Gaussian Process (GP) surrogate model is fitted to the observed data  $\mathcal{D}_n$ , capturing our current understanding of the objective function. Using this surrogate model, the algorithm maximizes the acquisition function to find the next promising set of hyperparameters:

$$\mathbf{x}_{n+1} = \arg \max_{\mathbf{x} \in \mathcal{X}} \alpha(\mathbf{x}; \mathcal{D}_n). \quad (11)$$



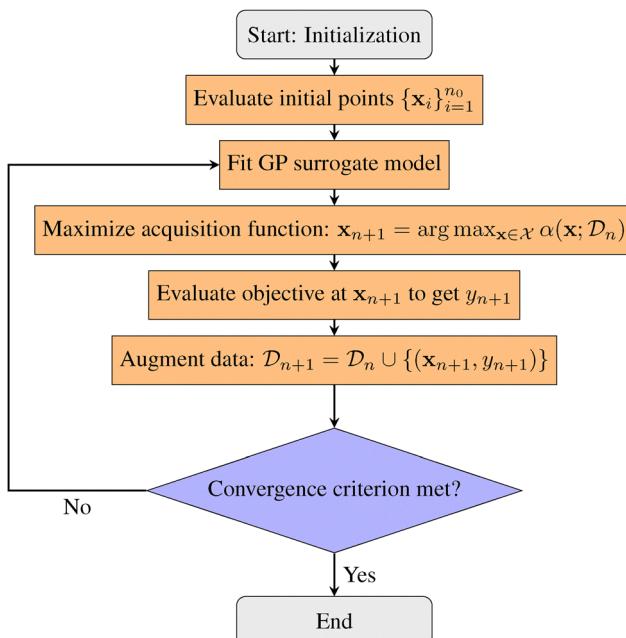


Fig. 7 Flowchart of the Bayesian optimization loop. The process iteratively updates the surrogate model and selects new hyperparameters to evaluate until convergence criteria are met.

The objective function is then evaluated at  $x_{n+1}$  to obtain  $y_{n+1}$ , and the data is augmented:

$$D_{n+1} = D_n \cup \{(x_{n+1}, y_{n+1})\}. \quad (12)$$

The algorithm checks if the convergence criterion is met (e.g., maximum iterations reached or negligible improvement observed). If not, the process loops back to updating the surrogate model with the new data, as depicted in Fig. 7. This iterative process continues until convergence, resulting in the optimal set of hyperparameters.

**2.4.6 Parameter space.** We optimized 13 hyperparameters of the image processing algorithm, each within specified bounds informed by prior knowledge as shown in Table 2. The ranges were chosen to cover a wide search space while ensuring that the parameters were physically meaningful and relevant to the task of detecting crystalline regions in HRTEM images.

Table 2 Hyperparameters and their ranges used in Bayesian optimization

Hyperparameter	Range	Type
blur_iteration	[5, 20]	Integer
Blur_kernel_propCons	[0.1, 0.5]	Real
closing_k_size	[1, 20]	Integer
opening_k_size	[1, 20]	Integer
pixThresh_propCons	[0.0, 1.0]	Real
ellipse_len_propCons	[0.5, 5.0]	Real
ellipse_aspect_ratio	[2.0, 7.0]	Real
thresh_dist_propCons	[1.0, 5.0]	Real
thresh_theta	[5.0, 15.0]	Real
cluster_size	[1, 10]	Integer
dspace_bandpass	[0.1, 0.5]	Real
powSpec_peak_thresh	[1.0, 1.5]	Real
thresh_area_factor	[1.0, 5.0]	Real

**2.4.7 Objective function evaluation.** Fig. 8 shows example inputs for the Bayesian optimization training. We used a total of 13 annotated HRTEM images for training. The ground truth annotations are created using VGG annotator tool.<sup>37</sup> For each set of hyperparameters  $\mathbf{x}$ , the objective function  $f(\mathbf{x})$  was evaluated as follows:

1. Run the image processing algorithm with parameters  $\mathbf{x}$  on the annotated HRTEM images.

2. Generate binary masks of the detected crystalline regions.

3. Compute the IoU between the detected masks and the ground truth masks:

$$\text{IoU}(\mathbf{x}) = \frac{|\text{Detected} \cap \text{Ground Truth}|}{|\text{Detected} \cup \text{Ground Truth}|}, \quad (13)$$

where  $|\cdot|$  denotes the cardinality (number of pixels in this context).

4. Set the objective function value as  $f(\mathbf{x}) = -\text{IoU}(\mathbf{x})$ .

#### 2.4.8 Training-set selection and annotation effort.

GRATEv2 is tuned per dataset; a single global parameter file is unlikely to generalise across different polymer systems or imaging conditions for three reasons:

1. Scale dependence. Several thresholds (e.g. backbone length, bone length, adjacency distance) are expressed as multiples of the lattice  $d$ -spacing and therefore change when the spacing or pixel size differs from one experiment to another.

2. Contrast variation. Grey-level statistics vary with beam current, detector settings and sample preparation, affecting Otsu thresholding and morphological noise removal.

3. Morphology differences. Polymer batches can exhibit distinct degrees of lamellar overlap, curvature and fringe density, all of which influence optimal parameter values.

From 653 HRTEM micrographs we hand-selected 13 images ( $\approx 2\%$ ) that together span the extremes of contrast, crystal size, overlap and orientation visible in the full batch. Each image was annotated with the VGG Image Annotator;<sup>37</sup> annotating one image takes about 5 minutes, so the entire training set required  $\sim 1$  hour of manual effort.

We recommend the following protocol as a practical guide for selecting a training set and annotating images for GRATEv2:

1. Randomly draw 10–15 images, then replace or add a few so that the set covers the darkest, brightest, most densely fringed and sparsest cases in the batch.

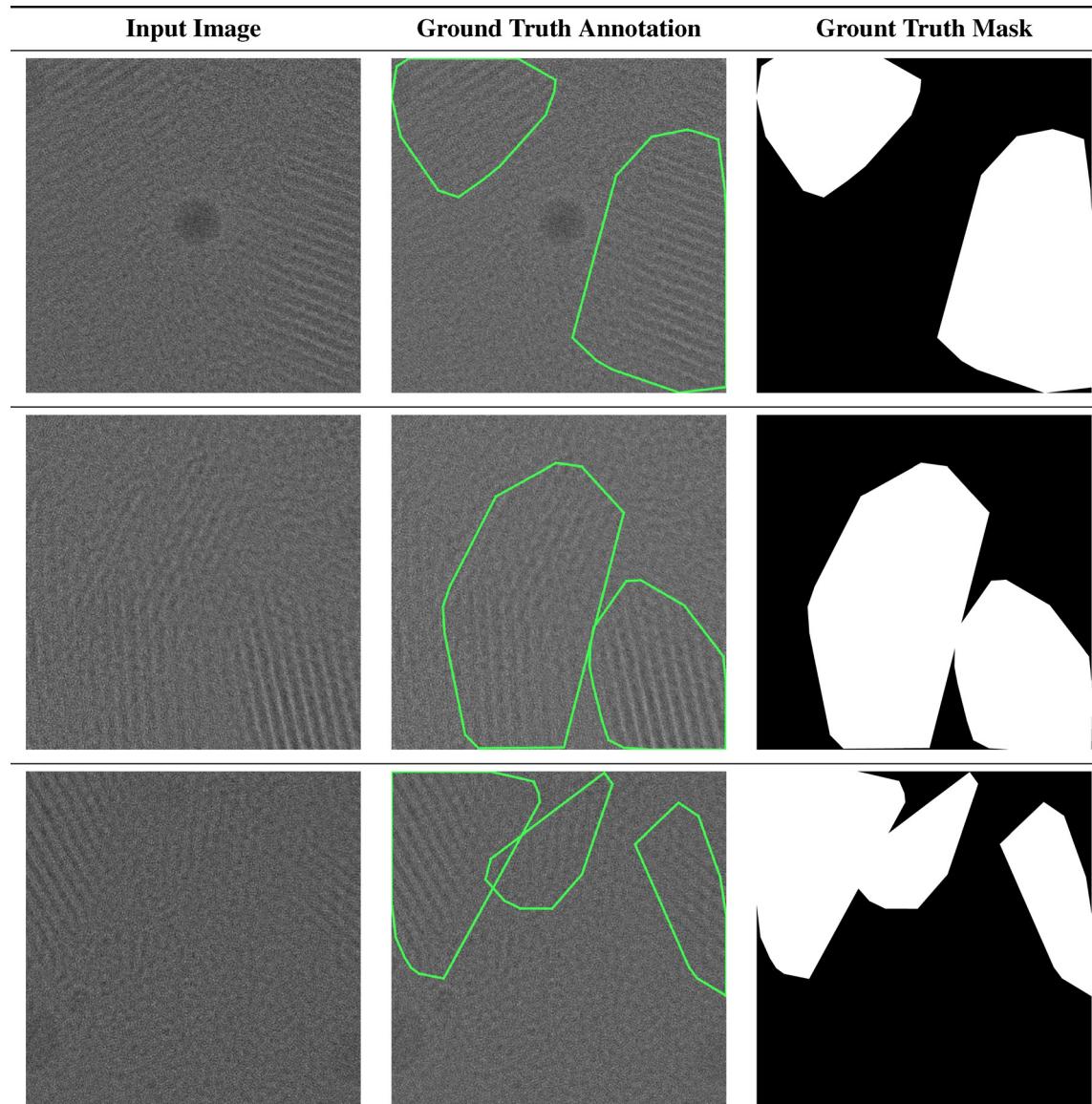
2. Annotate only this subset; run Bayesian optimisation once.

3. Apply the resulting parameter file to the remainder of the dataset.

Linking parameter selection to just 13 annotated images therefore keeps manual effort to about one hour while enabling automated analysis of the entire dataset.

**2.4.9 Algorithm execution.** Our implementation of the Gaussian Process-based Bayesian optimization algorithm for hyperparameter tuning is based on the scikit-optimize library.<sup>38</sup> The library provides a user-friendly interface through the `gp_minimize` function for Bayesian optimization and supports various acquisition functions, surrogate models, and optimization strategies.

We used the `gp_minimize` function with the following key parameters:



**Fig. 8** Bayesian optimization training input *i.e.* input images, ground truth annotations, and ground truth masks. Each row represents a different sample from the dataset. 13 manually annotated images are used for training in this work.

- Acquisition function: expected improvement (EI).
- Number of calls: 200 total evaluations of the objective function.
- Initial points: 10 random initial evaluations to seed the GP model.
- Random state: seeded for reproducibility.

## 2.5 Evaluation of data sufficiency

Determining the optimal amount of data to collect is crucial for experimentalists. Collecting too little data can compromise the reliability of results, while collecting excessive data may not yield additional insights and can waste valuable resources. Establishing a reliable stopping criterion for data collection ensures that resources are utilized efficiently without sacrificing statistical significance.

An effective data sufficiency metric should possess certain desirable features:

1. Sensitivity to distribution changes: it should accurately reflect changes in the underlying data distribution as more data is collected.

2. Scale interpretability: the metric should provide a quantitative measure that is interpretable and can be related to practical thresholds for decision-making.

3. Applicability to empirical distributions: it should be suitable for comparing empirical distributions derived from finite samples.

4. Metric properties: the measure should satisfy the properties of a mathematical metric, such as non-negativity, identity of indiscernibles, symmetry, and triangle inequality, to ensure consistent and meaningful comparisons.

In this study, we introduce a stopping criterion based on the Wasserstein distance to assess data sufficiency. The Wasserstein distance, also known as the Earth Mover's Distance, quantifies the difference between two probability distributions



by measuring the minimum “cost” of transforming one distribution into the other. It is particularly well-suited for our purposes because it satisfies all the desired features of a data sufficiency metric listed above.

### 2.5.1 Mathematical definition of the Wasserstein distance.

Let  $P$  and  $Q$  be two probability distributions on the real line with cumulative distribution functions (CDFs)  $F_P(x)$  and  $F_Q(x)$ , respectively. The  $p$ -th order Wasserstein distance  $W_p(P, Q)$  between  $P$  and  $Q$  is defined as:

$$W_p(P, Q) = \left( \int_{-\infty}^{\infty} |F_P^{-1}(u) - F_Q^{-1}(u)|^p du \right)^{1/p} \quad (14)$$

where  $F_P^{-1}(u)$  and  $F_Q^{-1}(u)$  are the quantile functions (inverse CDFs) of  $P$  and  $Q$ , and  $p \geq 1$ . For  $p = 1$ , the first-order Wasserstein distance simplifies to:

$$W_1(P, Q) = \int_0^1 |F_P^{-1}(u) - F_Q^{-1}(u)| du \quad (15)$$

Alternatively, for discrete empirical distributions derived from finite samples, the first-order Wasserstein distance between two sets of observations  $\{x_i\}_{i=1}^n$  and  $\{y_j\}_{j=1}^m$  can be computed by sorting the observations and calculating the average absolute difference between the sorted values:

$$W_1(P_n, Q_m) = \frac{1}{N} \sum_{k=1}^N |x_{(k)} - y_{(k)}| \quad (16)$$

where  $N = \min(n, m)$ , and  $x_{(k)}, y_{(k)}$  are the ordered statistics (sorted data).

The Wasserstein distance’s ability to capture differences between distributions, even when they have overlapping support, makes it ideal for assessing the convergence of empirical data distributions as more data is collected. By monitoring the Wasserstein distance between successive data samples, we can determine when the distributions have converged, indicating that the data collection process has reached a point of diminishing returns.

## 3 Results

We first provide brief details of the material and process used to collect the HRTEM data for completeness. To prepare TEM samples, 5 mg mL<sup>-1</sup> solutions of PCDTBT were dissolved in chlorobenzene within a nitrogen glovebox at 45 °C for at least 12 hours. Silicon wafers were cleaned by sonication for 20 minutes in acetone, followed by 20 minutes in isopropanol, and subsequently subjected to UV-ozonation for 20 minutes. Poly(3,4-ethylenedioxythiophene)-poly(styrene sulfonate) (PEDOT:PSS) films were cast onto silicon substrates by spin coating at 4000 rpm for 2 minutes in air to serve as a sacrificial layer, facilitating film floating. Substrates were then transferred to a nitrogen glovebox, where PCDTBT films were spin-cast at 800 rpm for 2 minutes. For TEM sample preparation, the coated substrates were removed from the glovebox, floated in deionized water, and carefully transferred onto copper TEM grids. These samples were left under ambient conditions to dry overnight and then annealed

in the nitrogen glovebox at 190 °C for 2 hours. High-resolution TEM (HRTEM) imaging was performed at the Penn State Materials Characterization Lab using the FEI Titan Krios microscope operating at 300 kV, equipped with the K2 direct electron detector and a cryo-stage. A dose rate of 20 e Å<sup>-2</sup> s was used for a 2.5 s exposure. An automated acquisition process was set to autofocus before each capture at 300k $\times$  magnification, with a randomly assigned defocus value between 0 and -3 μm. Images were acquired at 470k $\times$  magnification with a 2.5 μm step size between exposed regions, resulting in a total of 637 images for subsequent analysis.

Using GRATEv2, we detected crystals within each of the HRTEM images. For each identified crystal, the following features were extracted (see Section 3.2): center-of-mass coordinates, orientation angle relative to the image axis,  $d$ -spacing, and crystal lengths along both the major and minor axes. Through this process, GRATEv2 identified a total of 4350 ordered domains from the HRTEM images. In Section 3.3, we present the timing performance of GRATEv2, along with the time distribution among different components of the analysis process. In Section 3.4, we evaluate the data sufficiency—that is, how many images are sufficient to achieve statistical convergence in the extracted features.

### 3.1 Advantages of Bayesian parameter optimization

To evaluate the benefit of adding Bayesian optimisation (BO) to GRATEv2, we compare exactly the same rule-based pipeline under two hyper-parameter sets: (i) manually selected after extended trial-and-error, and (ii) automatically selected by BO. Performance is measured with the pixel-wise Intersection-over-Union (IoU) against expert masks on six representative HRTEM images. Absolute IoU values vary with crystal morphology and noise, so we emphasise the relative gain and supply visual examples so that readers can see what IoU  $\simeq 0.43$  and IoU  $\simeq 0.58$  actually look like.

**3.1.1 Quantitative comparison of IoU scores.** Table 3 presents the IoU scores for each image using both the manually selected parameters and the Bayesian-optimized parameters. Additionally, the table includes the differences in IoU scores between the two parameter sets for each image. The average and standard deviation of IoU score across the six images is also calculated for both cases.

The average IoU score using the Bayesian-optimized parameters is 0.5784, representing an improvement of approximately 32.3% over the average IoU of 0.4371 obtained with the manually selected parameters. This significant increase demonstrates the effectiveness of Bayesian optimization in enhancing the algorithm’s performance in detecting and segmenting crystalline regions in HRTEM images. The computed  $t$ -value corresponding to  $t$ -statistic is 7.074, which exceeds the critical  $t$ -value of 2.571, indicating strong statistical justification for improvement.

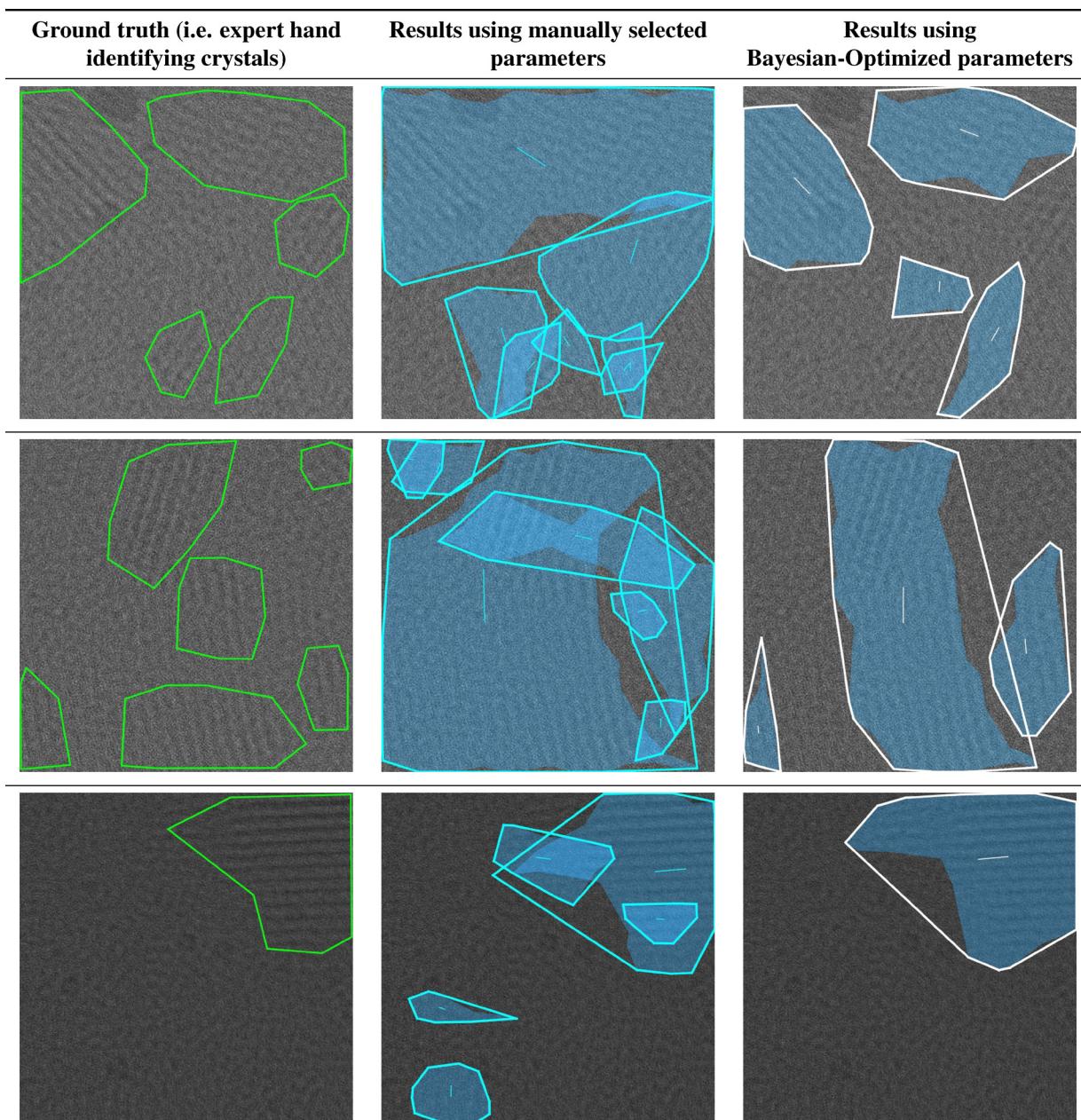
**3.1.2 Visual comparison of detection results.** Numbers alone can be abstract. Fig. 9 and Appendix C pairs each IoU with its mask so the reader can judge what a pixel-wise IoU of 0.43 (manual) *versus* 0.58 (BO) means in practice. Visual



**Table 3** IoU scores over validation dataset for manual and Bayesian-optimized parameters with differences

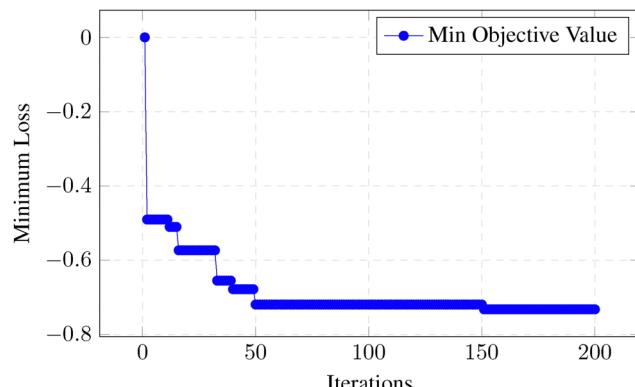
Image filename	Manual selected parameters IoU	Bayesian parameters IoU	IoU difference, $d_i$
1.tif	0.5296	0.6911	+0.1615
2.tif	0.3274	0.5156	+0.1882
3.tif	0.3178	0.4934	+0.1756
4.tif	0.5285	0.5864	+0.0579
5.tif	0.3906	0.5400	+0.1494
6.tif	0.5288	0.6438	+0.1150
<b>Average IoU</b>	<b>0.4371</b>	<b>0.5784</b>	<b>+0.1413</b>

comparisons illustrate that the algorithm using Bayesian-optimized parameters generally provides a more accurate and comprehensive detection of crystalline regions than the manually tuned approach. The Bayesian-optimized parameters tend to align the detected crystal boundaries more consistently with the annotated features. In particular, the algorithm with Bayesian-optimized parameters appears to capture finer details and produce clearer delineation of crystal boundaries, suggesting a closer correspondence to the expert annotations than that achieved under manual parameter tuning.



**Fig. 9** Comparison of ground truth, manually selected parameter detection, and Bayesian-optimized parameter detection across three different images. Each column represents a distinct detection method, illustrating how Bayesian optimization enhances segmentation accuracy by more closely matching the ground truth annotations compared to manual parameter tuning. The Bayesian optimization process was conducted over 200 iterations, achieving a minimum loss value of  $-0.7319$  at the 151st evaluation.





**Fig. 10** Convergence of the Bayesian optimization process over 200 iterations, illustrating the reduction in the loss function (negative IoU) as optimization progresses. The y-axis represents the minimum loss value achieved up to that evaluation. The minimum loss value of  $-0.7319$  was attained at the 151st evaluation.

### 3.1.3 Convergence of the Bayesian optimization process.

Fig. 10 depicts the convergence of the Bayesian optimization process. The objective function, defined as the negative mean IoU, decreases over successive iterations, indicating that the optimization algorithm effectively identifies hyperparameters that enhance the segmentation performance. The convergence plot demonstrates that significant improvements are achieved within the first 51 iterations, after which the objective function gradually approaches a plateau.

### 3.1.4 Analysis of optimized parameters.

Appendix A compares the values of the manually selected parameters with those obtained through Bayesian optimization.

Several notable differences are observed between the parameter sets:

- Morphological operations: the Bayesian-optimized parameters for closing\_k size and opening\_k size are significantly smaller than the manual values (2 vs. 15 and 17, respectively). This suggests that less aggressive morphological operations

preserve finer details in the images, contributing to improved segmentation accuracy.

- Edge detection and filtering: parameters like ellipse\_len\_propCons and dspace\_bandpass are adjusted to better capture the characteristics of the crystalline structures. The increase in ellipse\_len\_propCons from 1.5 to 4.03 indicates a preference for detecting longer ellipses, aligning with the elongated shapes of crystals.

- Thresholding parameters: the pixThresh\_propCons is slightly higher in the Bayesian-optimized parameters, which may help in differentiating crystals from the background noise more effectively.

### 3.1.5 Advantages of Bayesian optimization.

The integration of Bayesian optimization into our image processing framework offers several advantages:

- Enhanced performance: the Bayesian-optimized parameters significantly improve the mean IoU score by approximately 32.3%, indicating superior segmentation performance.

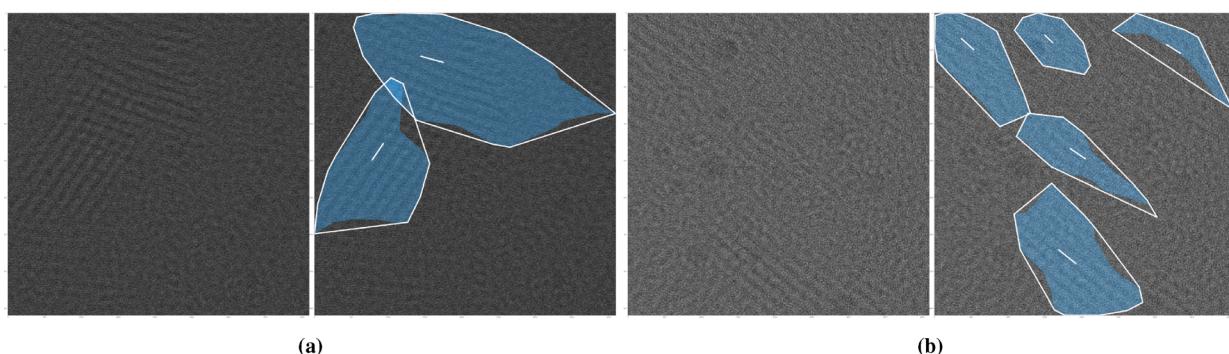
- Automated tuning: Bayesian optimization automates the hyperparameter tuning process, reducing the need for manual intervention and deep image-processing expertise, thereby saving time and resources. This is especially important when the number of hyperparameters (here, 13) makes manual exploration rather tedious.

- Efficient exploration: the optimization process efficiently explores the hyperparameter space, converging to optimal values in fewer evaluations compared to exhaustive search methods. This is evident from the convergence plot in Fig. 10.

- Robustness: the optimized parameters produce robust results across various images, as evidenced by consistent improvements in IoU scores.

## 3.2 Structural feature extraction of crystals from HRTEM images

Using the parameter set listed in Appendix A, we applied GRATEv2 to detect and analyze crystalline domains within the HRTEM images of PCDTBT. The segmentation results for two images are presented in Fig. 11, where the identified crystals



**Fig. 11** (a) and (b) Are 1.tif and 2.tif images respectively corresponding to Tables 4 and 5 (a) and (b) shows the original image (left) and the segmentation output (right) from our algorithm for HRTEM of PCDTBT. The detected crystals have a  $d$ -spacing of 1.9 nm. The image on the left is the input to the algorithm, and on the right is the output of the algorithm. Each detected crystal in the output shows (1) the convex hull boundary around the crystal, (2) the shaded region representing a more exact crystal region, and (3) a straight line at the centroid of the convex hull, which shows the orientation of the crystal patterns.



Table 4 Features of the detected crystals shown in Fig. 11

Name	Centroid ( $p_x, p_y$ )	Area (nm $^2$ )	Angle (deg)	<i>d</i> -Spacing (nm)	MajorAxis (nm)	MinorAxis (nm)	AxisAngle (deg)
1.tif	(1748, 670)	589.7	-164.7	2.1	21.1	10.2	23.8
	(785, 1992)	293.9	-55.9	2.0	14.8	9.9	-65.5
2.tif	(534, 497)	177.9	-137.9	1.9	12.5	5.3	53.0
	(1607, 402)	84.3	-136.0	0.8	7.9	4.3	35.4
	(3345, 546)	71.7	-148.8	1.7	14.4	4.8	39.6
	(2050, 1975)	125.4	-146.0	2.2	16.0	4.2	34.1
	(1922, 3396)	263.8	-141.4	2.0	14.2	8.0	46.9

are highlighted. Each detected crystal is surrounded by a convex hull boundary, with a shaded region representing a more precise delineation of the crystallite. A line at the crystal centroid indicates its orientation. The extracted properties for these detected crystals are summarized in Table 4, including their centroid coordinates, area, orientation angle, *d*-spacing, major and minor axis lengths, and axis angle. Additionally, the correlation measurements between pairs of crystals are provided in Table 5, featuring metrics such as metric distance, direct distance, and relative angle.

The detection of these crystalline domains allows for a comprehensive analysis of the microstructural features of PCDTBT. The *d*-spacing values shown in Fig. 12b range from approximately 1.1 nm to 2.9 nm, which are consistent with the expected lattice spacings associated with PCDTBT's crystalline structures.<sup>39</sup> Variations in *d*-spacing among the detected crystals may be attributed to differences in crystallite orientation, strain within the material, or inherent structural disorder due to the semi-crystalline nature of PCDTBT.

The areas of the detected crystals vary significantly as shown in Fig. 12a, with values ranging from approximately 14.89 nm $^2$  to 2307.18 nm $^2$ . Larger crystal areas may correlate with improved charge transport properties, as larger crystalline domains can facilitate more efficient charge carrier mobility along the polymer chains.<sup>27</sup> The aspect ratios, derived from the major and minor axis lengths, provide insights into the shapes of the crystals. Higher aspect ratios indicate elongated, rod-like crystals, while lower aspect ratios suggest more equiaxed or spherical shapes. The diversity in crystal shapes and sizes can impact the overall morphology and performance of the polymer in electronic applications.

Fig. 12 also presents additional statistical analyses of individual crystal properties across the dataset, including histograms of crystal areas, *d*-spacings, orientation angles, and shape descriptors like aspect ratio. These plots allow us to identify prevalent features and distributions within the material. For example, the

histogram of *d*-spacings may show a peak around a specific value, indicating a dominant crystalline phase or preferred stacking distance within the polymer chains.

The comprehensive analysis provided by GRATEv2 enables us to establish quantitative relationships between microstructural features and potential material performance. By systematically characterizing the size, shape, orientation, and spatial distribution of crystalline domains, we can correlate these features with electronic properties measured in devices. This level of detailed microstructural understanding is essential for guiding the design and processing of conjugated polymers to achieve optimal performance in organic electronic applications.

For a more extensive and detailed exploration of this PCDTBT dataset, including insights into intercrystalline correlations and preferred crystallographic alignments, readers are referred to the work of our collaborators in ref. 40. Their study leverages automated HRTEM and the GRATEv2 image processing outputs to unravel how neighboring crystals preferentially align along certain lattice directions, likely reflecting underlying liquid crystalline order within the polymer. By combining the analysis presented here with their comprehensive assessment of orientation correlations and lattice parameters, one obtains a richer and more complete understanding of the polymers nanoscale structure and its implications for organic electronics.

### 3.3 Timing statistics

The algorithm was executed on a computer equipped with a 96-core AMD EPYC 9654 CPU@3.7 GHz running Linux OS. The total time for processing a single high-resolution transmission electron microscopy (HRTEM) image with 1.9 nm *d*-spacing crystals is approximately 6.52 seconds when utilizing a single core. The timing consumption by various parts of the algorithm is presented in Fig. 13. The most time-consuming steps are skeletonization (approximately 4.66 seconds), followed by breaking branches and the preprocessing steps (blurring, histogram equalization, and thresholding). Utilizing all 96 cores, we processed an entire dataset of 637 images in 284 seconds (approximately 4 minutes and 44 seconds), reducing the per-image processing time to just 0.44 seconds. This significant improvement demonstrates the scalability and efficiency of our algorithm when parallelized across multiple cores.

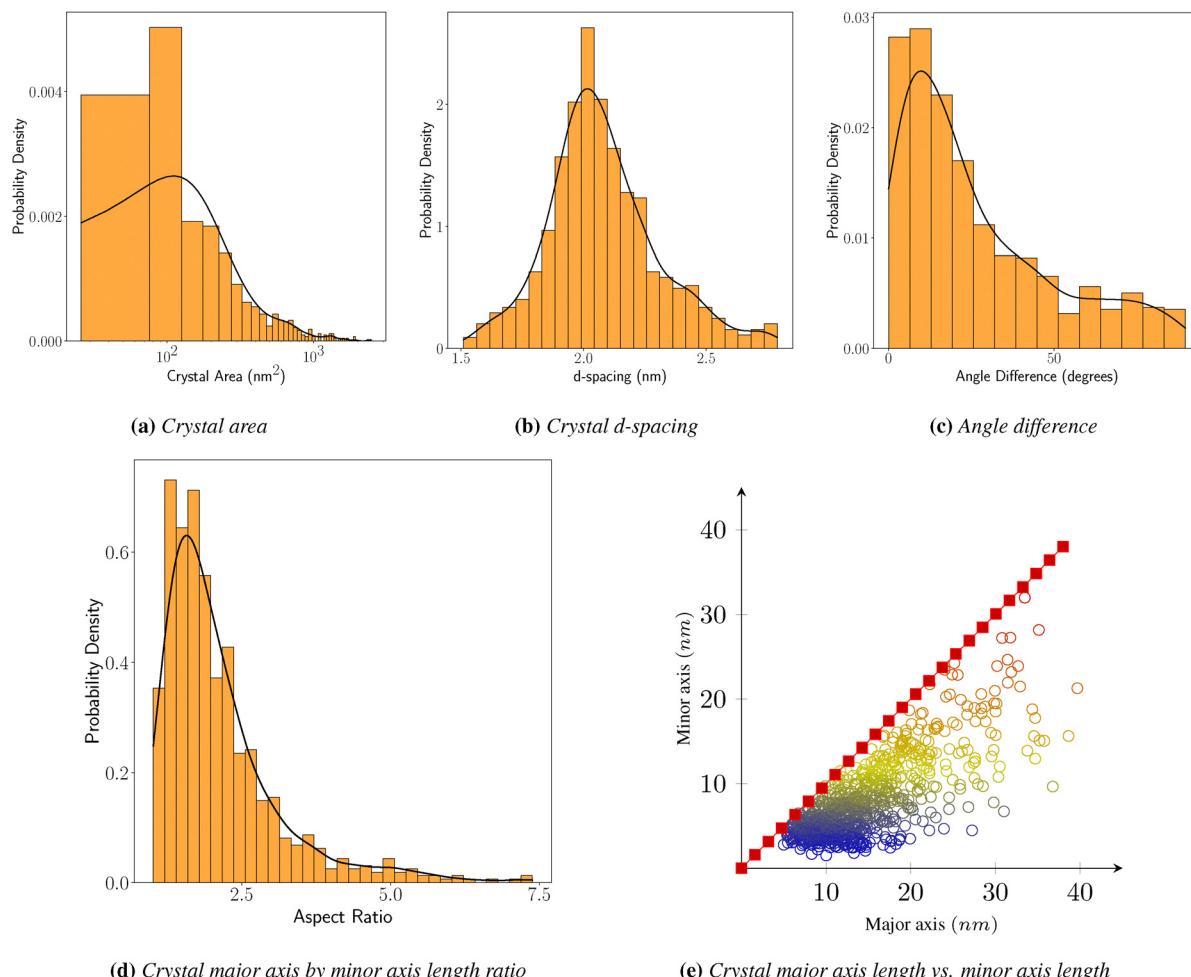
The performance enhancements of our algorithm are attributed to the use of optimized libraries and functions that efficiently handle computationally intensive tasks. Specifically, we employed:

1. The skeletonize function from skimage.morphology<sup>41</sup> for efficient skeletonization.

Table 5 Crystal correlation measurements for Fig. 11

Name	Metric distance (1)	Direct distance (nm)	Relative angle (deg)
1.tif	0.89	20.84	71.24
2.tif	2.91	35.81	10.95
	1.95	26.97	8.13
	2.45	40.94	3.5
	2.21	24.57	2.81
	2.91	40.58	7.45
	1.17	18.18	4.64





**Fig. 12** Individual crystal analysis. The properties of the individually detected crystals are plotted: (a) crystal area, (b) FFT evaluated *d*-spacing, (c) angle difference between crystal's pattern and its major axis, (d) crystal aspect ratio, and (e) crystal axis lengths. (a)–(d) Are histograms with kernel density estimate, and (e) is a scatter plot from analysis of the entire dataset.

2. OpenCV<sup>42</sup> functions `equalizeHist` and `threshold` (`cv2.equalizeHist`, `cv2.threshold`) for fast histogram equalization and image thresholding.
3. Morphological operations such as closing and opening using `cv2.morphologyEx` from OpenCV.
4. The `skeleton_to_csgraph` function from the `skan`<sup>43</sup> library for converting skeleton images to graph representations.
5. The `label` function from `skimage.measure`<sup>41</sup> for rapid image segmentation.
6. Sparse graph structures and connected component analysis using `csr_matrix` and `connected_components` from `scipy.sparse.csgraph`<sup>44</sup> for efficient graph construction and evaluation.
7. Fast Fourier transforms and other numerical operations using NumPy functions such as `np.fft`.<sup>45</sup>
8. The `alphashape` library to create shrink-wraps around point clouds.

By utilizing these optimized libraries, we minimized computational overhead and maximized the efficiency of each processing step. The skeletonization step, although still the

most time-consuming, benefits significantly from the optimized implementation in `skimage`.<sup>41</sup> Similarly, the use of sparse matrices and efficient graph algorithms from `scipy.sparse.csgraph`<sup>44</sup> greatly accelerates the analysis of the skeleton's connectivity.

Our analysis of the timing statistics reveals that the skeletonization step accounts for approximately 71% of the total processing time for a single image when executed on a single core. This indicates that skeletonization is a major computational bottleneck in the algorithm. However, due to the parallel nature of image processing tasks, distributing the workload across multiple cores significantly mitigates this bottleneck. By processing images concurrently, we effectively utilize the available computational resources, resulting in a substantial reduction in total processing time.

Furthermore, the efficient handling of large data structures, such as sparse matrices in graph construction and connected component analysis, contributes to the algorithm's scalability. The use of optimized libraries ensures that even computationally intensive tasks are executed as efficiently as possible. This



## Timing Statistics

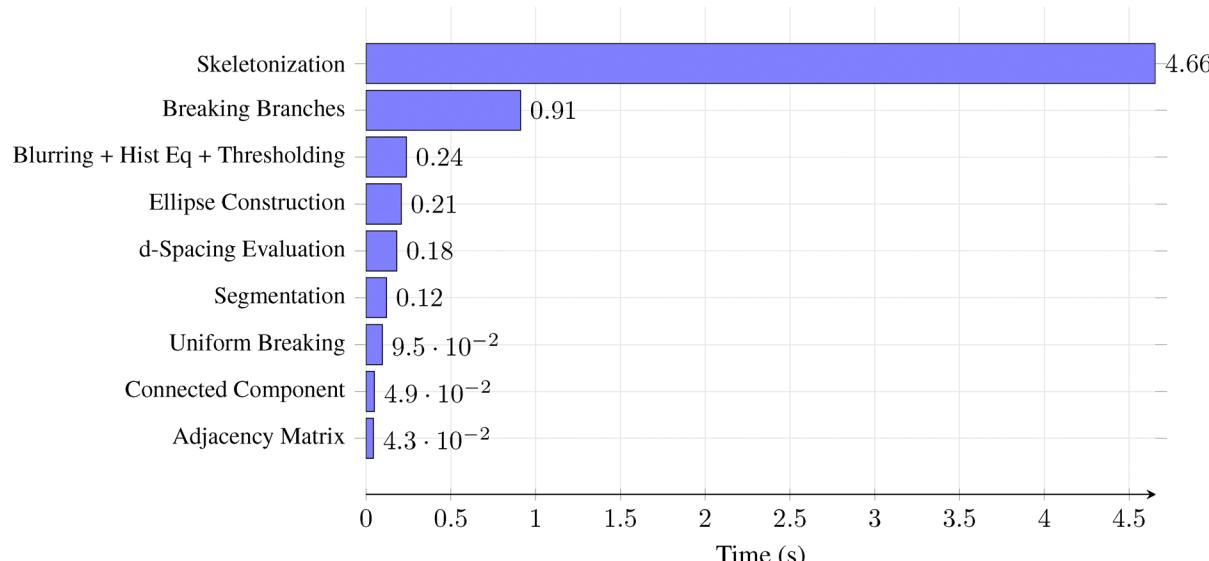


Fig. 13 Time taken by each step in the algorithm for the analysis of 1.9 nm  $d$ -spacing crystals in a single image using a single core. The total processing time is approximately 6.52 seconds.

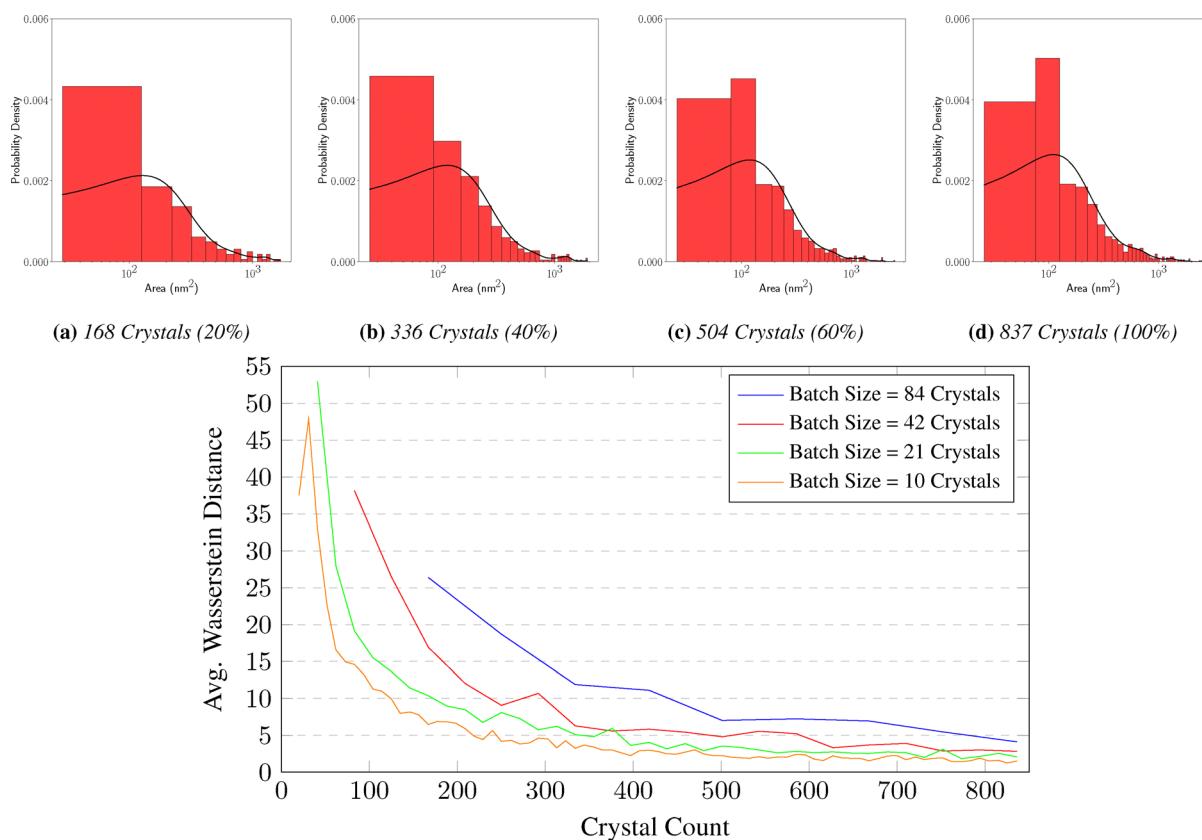


Fig. 14 Data sufficiency for the crystal-area feature. (a–d) Empirical area distributions (histogram + KDE) at cumulative sampling levels of 20% (168 crystals), 40% (336), 60% (504), and 100% (837). (e) Average Wasserstein distance between successive cumulative batches of crystal areas versus crystal count; curves indicate batch sizes: 84 (blue), 42 (red), 21 (green), and 10 (orange). Lower values indicate distributional convergence.



optimization is crucial when dealing with large datasets, as it enables rapid analysis without compromising accuracy or resolution.

The combination of algorithmic efficiency, optimized library functions, and effective parallelization allows our method to achieve high performance in processing and analyzing large volumes of HRTEM images. This capability is essential for applications requiring rapid data analysis and real-time feedback in materials science and related fields.

Preparing ground-truth masks for 13 representative images required about 5 minutes per image ( $\approx 1$  h total hands-on time). Bayesian optimisation of the 13-dimensional parameter space then ran unattended for 200 iterations, taking  $\approx 90$  min on a single HPC node, after which the resulting parameter file was applied to the remaining 640 images. In practical terms, one hour of annotation replaced the  $\gtrsim 60$  h of manual trial-and-error previously needed to tune the parameters manually. The Bayesian optimisation process was able to quickly identify the optimal parameter set that maximized the segmentation performance across the training dataset, demonstrating the effectiveness of this approach in reducing expert effort while maintaining high accuracy.

### 3.4 Data sufficiency analysis of PCDTBT crystals in HRTEM images

In our analysis of data sufficiency, we examine how the distribution of crystal areas reaches an asymptotic distribution as we incorporate more HRTEM images of PCDTBT. The complete dataset consists of 837 crystals from around 600 images. To evaluate how quickly the underlying distribution of crystal areas converges, we incrementally add data in fixed-size batches and compute the first-order Wasserstein distance  $W_1$  between distributions formed by consecutive increments.

Specifically, we consider four different batch sizes at which images (rather, crystals) are added to the dataset: 10, 21, 42, and 84 crystals. For each batch size scenario, we start from 0 crystals and progressively add data in increments equal to the batch size until we reach the full dataset size (of 837 crystals). After each increment, we compute the Wasserstein distance between the new distribution (including  $i \times \text{batchSize}$  crystals) and the previous distribution (with  $(i - 1) \times \text{batchSize}$  crystals). To obtain a more stable and reliable statistic, we repeat this evaluation 10 times at each increment, each time randomly sampling the  $(i - 1)$  batches from  $i$  batches of crystals and then average the resulting distances.

**Table 6** Wasserstein distance computed on the crystal-area distribution: full dataset *versus* the distribution obtained after removing a single batch of images, evaluated for several batch sizes

Batch size	Wasserstein distance (full <i>vs.</i> one batch less)
84	4.12
42	2.83
21	2.08
10	1.51

As shown in Fig. 14 the histograms become smoother and more stable as the dataset grows, demonstrating that the distribution of crystal areas converges toward a steady form. Meanwhile, Fig. 14e shows how the averaged Wasserstein distance between consecutive increments decreases with additional data. Each curve represents a different batch size, revealing that the scale of incremental changes to the distribution depends on how many crystals are added at once.

Larger batch sizes (*e.g.*, 84 crystals) introduce more data at each increment, leading to more pronounced changes in the distribution per step. Once enough large increments have been included, the distribution may show a sudden, relatively large drop in Wasserstein distance, then rapidly converge. In contrast, smaller batch sizes (*e.g.*, 10 crystals) add data more gradually, producing smoother and more frequent updates. Each small increment makes a subtler change to the distribution, resulting in a more gradual and fine-grained trajectory toward convergence.

Because each batch size scenario scales the increments differently, the threshold for deciding when to stop data collection should also be scaled accordingly. For larger batch sizes, even a modest Wasserstein distance value (*e.g.*, around 4 units for the 84-crystal batch) might indicate sufficient convergence, since one large increment can naturally shift the distribution more. For smaller batch sizes, where each increment is gentler (*e.g.*, around 1.5 units difference for the 10-crystal batch), a lower threshold might be more appropriate, reflecting the finer control and resolution over the distributions shape. To illustrate this, Table 6 shows the Wasserstein distance between the full dataset and a dataset that is one batch smaller, for each considered batch size scenario. These values provide concrete examples of how batch size influences the scale of change in the distribution.

With this information, an experimentalist might set a higher threshold for larger batch sizes (*e.g.*, around 4 units for an 84-crystal batch) and a lower threshold for smaller batch sizes (*e.g.*, closer to 1.5 units for a 10-crystal batch). This ensures that the threshold for deciding when to cease data collection remains proportionate to the scale of changes induced by each incremental addition of data.

In practical terms, experimentalists can use insights from Fig. 14e and Table 6 to tailor their data collection strategy. By selecting an appropriate batch size and a corresponding Wasserstein distance threshold, they can decide when further data provides diminishing returns. If quick feedback and high resolution of distribution changes are desired, a smaller batch size and a lower threshold can be chosen. If time or resources are limited, larger batch sizes and a slightly higher threshold may be more suitable. In either case, once the averaged Wasserstein distance consistently falls below the chosen threshold, the experimentalist can confidently cease data collection, knowing the distribution is sufficiently representative. This approach transforms data sufficiency from a guesswork exercise into a clear, data-driven criterion that guides experimental resource allocation and ensures that the resulting dataset meets the necessary statistical rigor. An experimentalist might:



- Begin data collection in batches: they determine a batch size (e.g., 10 crystals per batch) and start collecting HRTEM images in increments of that batch size.
- Periodic assessment: after each new batch of data, they compute the Wasserstein distance between the current and previous distributions of crystal areas. This computation can be done after every increment, providing immediate feedback on the impact of newly acquired data.
- Decision point: if, after a certain number of increments, the averaged Wasserstein distance consistently falls below the established threshold (e.g., 1.5 units), the experimentalist has quantitative evidence that adding more data is unlikely to yield new insights into the crystal area distribution.
- Stopping data collection: with this statistical criterion, the experimentalist can confidently stop collecting further HRTEM images, reallocating their time and resources. This prevents unnecessary prolonged imaging campaigns.

## 4 Conclusions

In this work, we have developed and presented GRATEv2, an open-source computational framework for the automated analysis of high-resolution transmission electron microscopy (HRTEM) images, specifically focusing on complex microstructures in conjugated polymers like PCDTBT. By leveraging fast, automated image processing algorithms augmented with Gaussian process optimization, GRATEv2 significantly reduces the need for manual selection of parameters and tuning, enhancing both reproducibility and user accessibility. The integration of a Wasserstein distance-based stopping criterion within GRATEv2 provides a quantitative method for optimizing data collection, ensuring efficient use of transmission electron microscopy (TEM) resources without compromising data quality.

GRATEv2's compatibility with HPC environments allows for efficient, large-scale data processing at near real-time speeds, making it suitable for high-throughput applications in materials science. By successfully applying GRATEv2 to a substantial PCDTBT dataset, we demonstrated its efficacy in rapidly extracting critical structural features such as *d*-spacing, orientation,

and shape metrics. This capability is particularly valuable for advancing research in organic electronics, where precise nanoscale characterization is essential for optimizing material properties.

Overall, GRATEv2 addresses key limitations of existing HRTEM analysis methods by providing a fast, adaptable, and user-friendly tool that enhances the efficiency and reliability of microstructural characterization. By making GRATEv2 open-source, we aim to facilitate its adoption and further development by the research community. Future work could involve extending GRATEv2 to other material systems and incorporating additional analytical capabilities, thereby broadening its applicability and impact in the field of materials characterization.

## Code availability

The software developed for this paper is available at <https://github.com/baskargroup/GRATEv2>.

## Conflicts of interest

The authors have no competing interests to declare that are relevant to the content of this article.

## Data availability

The data and code for GrateV2 can be found at <https://github.com/baskargroup/GRATEv2>.

## Appendices

### A Optimum parameters

The Bayesian optimization based optimum parameters and manually selected parameters of GRATEv2 for our dataset are given in Table 7.

### B T-Statistic for data sufficiency

To evaluate the statistical significance of the improvement, we performed a paired *t*-test on the IoU scores from the two sets of

Table 7 Comparison of manually selected and Bayesian optimized parameters

Parameter	Description	Manually selected value	Bayesian optimized value
dspace_nm	<i>d</i> -Spacing in nm	1.9	1.9
pix_2_nm	Pixels per nm	78.5	78.5
blur_iteration	Blurring iterations	15	20
Blur_kernel_propCons	Kernel size blurring <sup>a</sup>	0.15	0.12
closing_k_size	Kernel size closing	15	2
opening_k_size	Kernel size opening	17	2
pixThresh_propCons	Threshold pixel length <sup>a</sup>	0.63	0.74
ellipse_len_propCons	Uniform breaking length <sup>a</sup>	1.50	4.03
ellipse_aspect_ratio	Threshold ellipse aspect ratio	5.00	4.38
thresh_dist_propCons	Adjacency distance <sup>a</sup>	2.00	1.36
thresh_theta	Adjacency angle (degrees)	10.00	13.96
cluster_size	Threshold cluster size	7	9
dspace_bandpass	Band pass filter size	0.20	0.44
powSpec_peak_thresh	Power spectrum threshold	1.15	1.00
thresh_area_factor	Area threshold factor	4.00	2.79

<sup>a</sup> Proportionality constant to *d*-spacing. dspace\_nm and pix\_2\_nm are user inputs and not optimized.



parameters. The null hypothesis ( $H_0$ ) is that there is no difference in the mean IoU scores between the manual and Bayesian-optimized parameters. Let  $d_i$  be the difference in IoU scores for each image, defined as  $d_i = \text{IoU}_{\text{Bayesian},i} - \text{IoU}_{\text{Manual},i}$ .

$$\bar{d} = \frac{1}{n} \sum_{i=1}^n d_i = 0.1413 \quad (17)$$

$$s_d = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (d_i - \bar{d})^2} = 0.0489 \quad (18)$$

The  $t$ -statistic is computed as:

$$t = \frac{\bar{d}}{s_d / \sqrt{n}} = \frac{0.1413}{0.0489 / \sqrt{6}} = \frac{0.1413}{0.01997} = 7.074. \quad (19)$$

with  $n - 1 = 5$  degrees of freedom, the critical  $t$ -value at a significance level of  $\alpha = 0.05$  (two-tailed) is approximately 2.571. Since  $t = 7.074 > 2.571$ , we reject the null hypothesis and conclude that the improvement in IoU scores using Bayesian optimization is statistically significant.

### C Additional validation results

Fig. 15.

### D Comparison with sliding-window FFT Benchmark

To contextualize the performance of GRATEv2, we implemented and tested a classical frequency-domain benchmark based on a sliding-window Fast Fourier Transform (SW-FFT) workflow, as is common for texture-based segmentation. This analysis reveals the

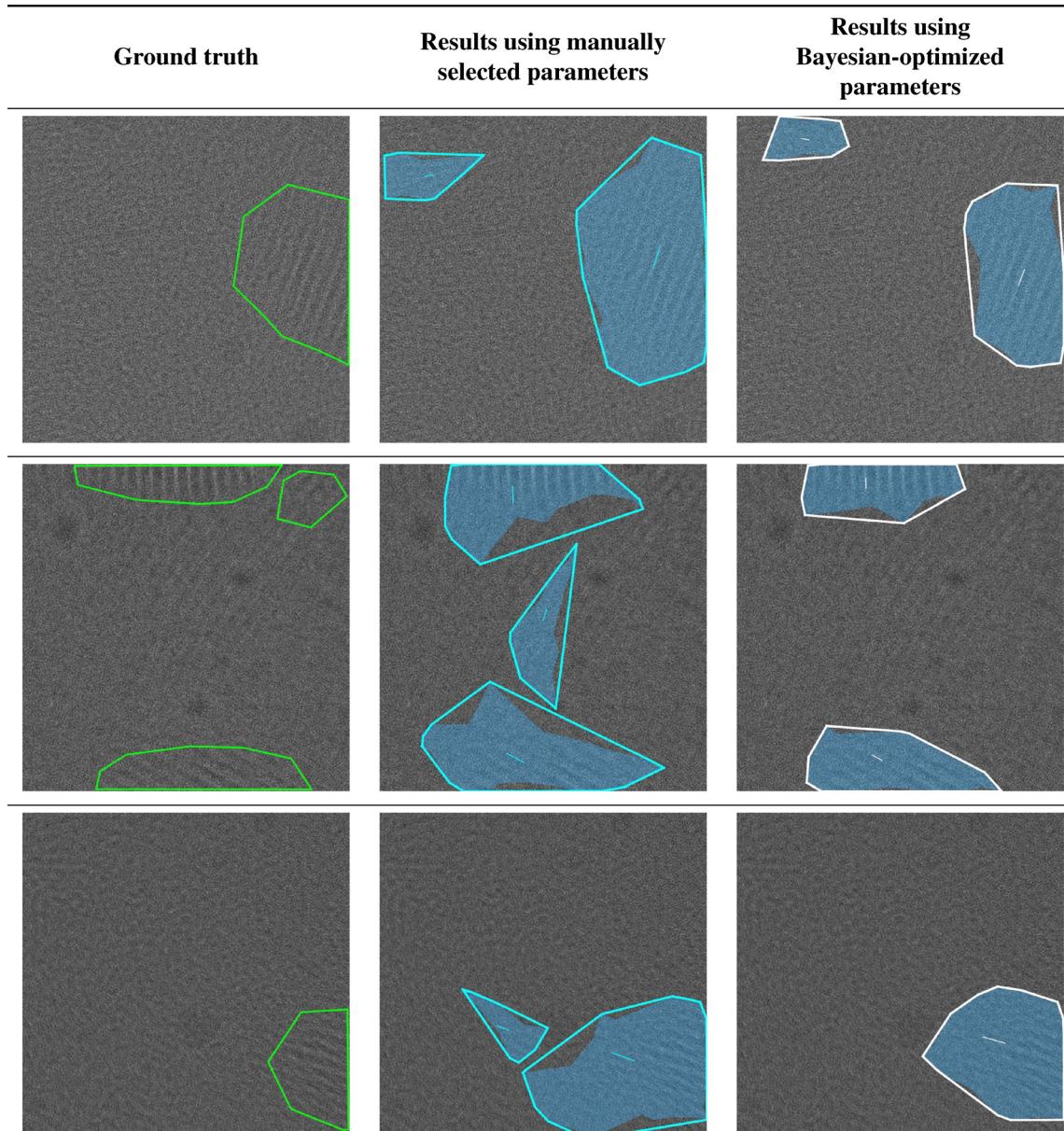


Fig. 15 Additional comparison of ground truth, manually selected parameters, and Bayesian-optimized parameters across different images.



limitations of such methods for our specific application and highlights the advantages of GRATEv2's graph-based spatial approach.

**SW-FFT methodology.** The SW-FFT algorithm was designed to identify crystalline regions based on their characteristic periodic fringes. The key steps are:

1. A square window of a fixed `window_size` slides across the image with a given stride.
2. For each window, a 2D-FFT is computed to generate a power spectrum.
3. A score is assigned to the window based on the maximum power spectral density within an annular (ring-shaped) frequency mask. This mask is defined by the expected *d*-spacing range of the material.

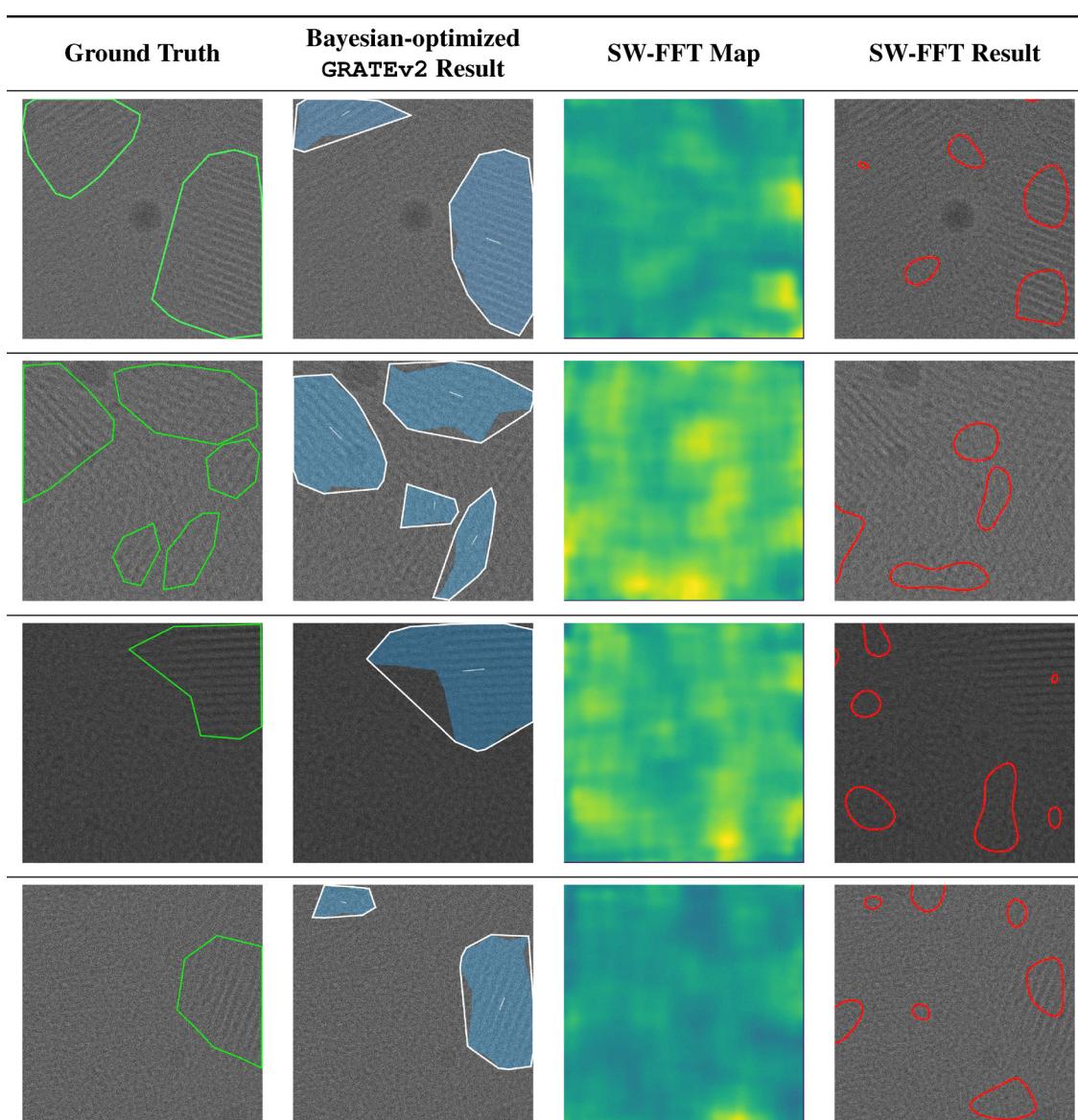
4. These scores are assembled into a 2D “crystallinity map,” which acts as a heatmap for crystalline likelihood.

5. This map is thresholded to produce a final binary segmentation mask of the detected crystalline domains.

The Python code for this implementation is provided in the github repository <https://github.com/baskargroup/GRATEv2>.

**Comparative results.** We applied the SW-FFT method to our PCDTB HRTEM dataset. Despite extensive parameter tuning (e.g., `window_size`, `stride`, `d_spacing_range`), the method failed to produce meaningful segmentation of the crystalline domains. The results are summarized in Fig. 16.

The primary findings are:



**Fig. 16** Visual comparison of segmentation results across multiple images. Column 1: Ground Truth HRTEM images. Column 2: Bayesian-optimized GRATEv2 performing well in segmenting crystalline domains, capturing their morphology and boundaries. Column 3: The crystallinity map from the SW-FFT method is noisy and lacks clear distinction. Column 4: The final segmentation from the SW-FFT method failing to capture the true crystal shapes, resulting in a poor mask that does not correspond to the actual crystalline domains. The SW-FFT results were generated using parameters: `window_size` = 640, `stride` = 16, and a wide `d_spacing_range` of (0.7, 3.7) nm and `threshold_quantile` = 0.90.



- Computational cost: the SW-FFT analysis on a single  $4096 \times 4096$  pixel image took approximately 12 min 17 s on average. In contrast, GRATEv2 processed the same image in 6.52 seconds on average, making it over 100 times faster. The high cost of the FFT method is due to the need to compute thousands of FFTs on large, overlapping windows.

- Segmentation quality: as visually demonstrated in Fig. 16, the SW-FFT method's qualitative performance is poor, failing to produce a meaningful or reliable segmentation mask. A direct comparison with the ground truth reveals several distinct and critical failure modes:

- Fragmentation and poor localization: the method struggles with spatial localization. Instead of identifying large, continuous crystals as single entities, it often detects them as a series of smaller, fragmented pieces concentrated in high-contrast areas. In contrast, GRATEv2 correctly captures the entire domain in one piece (Row 1). The SW-FFT method also completely ignored the large crystal in the top left of the image in Row 1, which is very well visible with the naked eye and precisely captured by GRATEv2.

- Noisy mapping and false detections: the generated crystallinity map (Column 3) is heavily diffused and noisy. This results in an unreliable segmentation that both completely misses major crystalline domains (false negatives) and incorrectly identifies numerous spurious regions where no crystals exist (false positives), as is evident in Rows 2 and 3.

- Poor correlation with ground truth: there is a fundamental disconnect between the high-score regions in the crystallinity map and the actual crystal locations. The algorithm often detects only a small, low-score portion of a true crystal while assigning the highest score to an entirely different, amorphous area. This poor correlation, combined with noise, leads to a fundamentally incorrect segmentation (Row 4).

This poor performance is not a matter of parameter choice but is a direct result of the method's core algorithmic flaw—the “Window Size Dilemma”.

- Need for a large window: to detect a periodic pattern *via* FFT, the analysis window must be large enough to contain multiple repetitions of the pattern. For our data, with a *d*-spacing of  $\approx 165$  pixels, a *window\_size* of at least 400–600 pixels is required. A smaller window sees only a fraction of a fringe, failing to register a periodic signal.

- Need for a small window: accurate segmentation requires high spatial precision to delineate the irregular boundaries of crystals. However, the SW-FFT method has a spatial resolution limited by its *window\_size*. Using a large window (*e.g.*,  $640 \times 640$ ) results in extremely poor localization, merging distinct nearby crystals and blurring boundaries with amorphous regions.

This required trade-off between pattern detection and spatial accuracy means no single *window\_size* can produce a satisfactory result. The issue is therefore an algorithmic flaw, not a matter of parameter tuning. While GRATEv2 benefits from Bayesian Optimization to navigate its complex 13-dimensional parameter space, the SW-FFT method's failure is due to these fundamental algorithmic limitations.

In contrast, GRATEv2's spatial-domain, graph-based approach is explicitly designed to overcome these challenges. By first identifying individual fringe-like skeletons and then evaluating their connectivity, it can robustly segment complex and irregularly shaped crystalline domains. This comparison therefore validates that classical frequency-domain methods are ill-suited for this class of material analysis, reinforcing the value and necessity of the specialized GRATEv2 framework.

## Acknowledgements

Funding from the National Science Foundation under Award DMREF 2323716 and the Office of Naval Research under Awards N00014-19-1-2453, and N00014-23-1-2001 are gratefully acknowledged.

## References

- 1 D. B. Williams and C. B. Carter, *Transmission Electron Microscopy: A Textbook for Materials Science*, Number v. 1 in Cambridge library collection. Springer, 2009. ISBN 9780387765006. URL <https://books.google.com/books?id=dXdrG39VtUoC>.
- 2 A. G. Cullis and P. A. Midgley, *Microscopy of Semiconducting Materials 2003*, CRC Press, 2018, ISBN 9781351083089. URL <https://books.google.com/books?id=5mUIEQAAQBAJ>.
- 3 R. F. Egerton, Electron energy-loss spectroscopy in the TEM, *Rep. Progress Phys.*, 2008, **72**(1), 016502, DOI: [10.1088/0034-4885/72/1/016502](https://doi.org/10.1088/0034-4885/72/1/016502).
- 4 J. C. H. Spence, *High-Resolution Electron Microscopy*, OUP Oxford, 2013. ISBN 9780191508400. URL <https://books.google.com/books?id=PitoAgAAQBAJ>.
- 5 T. Isabell, J. Brink, M. Kawasaki, B. Armbruster, I. Ishikawa, E. Okunishi, H. Sawada, Y. Okura, K. Yamazaki and T. Ishikawa, *et al.*, Development of a 200 kV atomic resolution analytical electron microscope, *Microsc. Today*, 2009, **170**(3), 8–11.
- 6 J. Kirkland, *Advanced Computing in Electron Microscopy*, Springer e-Books, Springer US, 2010. ISBN 9781441965332. URL <https://books.google.com/books?id=YscLlyaiNvoC>.
- 7 C. J. Brabec, A. Distler, X. Du, H.-J. Egelhaaf, J. Hauch, T. Heumueller and N. Li, Material strategies to accelerate OPV technology toward a GW technology, *Adv. Energy Mater.*, 2020, **10**(43), 2001864.
- 8 P. A. Midgley and R. E. Dunin-Borkowski, Electron tomography and holography in materials science, *Nat. Mater.*, 2009, **8**(4), 271–280.
- 9 P. Toth, A. B. Palotas, E. G. Eddings, R. T. Whitaker and J. S. Lighty, A novel framework for the quantitative analysis of high resolution transmission electron micrographs of soot *i.* improved measurement of interlayer spacing, *Combust. Flame*, 2013, **160**(5), 909–919. ISSN 0010-2180. URL <https://www.sciencedirect.com/science/article/pii/S0010218013000035>.
- 10 Y. Zhu, J. Ciston, B. Zheng, X. Miao, C. Czarnik, Y. Pan, R. Sougrat, Z. Lai, C.-E. Hsiung and K. Yao, *et al.*, Unravelling surface and interfacial structures of a metal-organic



framework by transmission electron microscopy, *Nat. Mater.*, 2017, **16**(5), 532–536.

- 11 A. Sharma, T. Kyotani and A. Tomita, A new quantitative approach for microstructural analysis of coal char using hrtem images, *Fuel*, 1999, **78**(10), 1203–1212.
- 12 P. Toth, Nanostructure quantification of turbostratic carbon by HRTEM image analysis: state of the art, biases, sensitivity and best practices, *Carbon*, 2021, **178**, 688–707.
- 13 B. Sesha Sarath Pokuri, J. Stimes, K. OHara, M. L. Chabinyc and B. Ganapathysubramanian, GRATE: a Framework and Software for Graph Based Analysis of Transmission Electron Microscopy Images of Polymer Films, *Comput. Mater. Sci.*, 2019, **163**, 1–10. ISSN 0927-0256. URL <https://www.sciencedirect.com/science/article/pii/S0927025619301041>.
- 14 U. Pratiush, A. Houston, S. V. Kalinin and G. Duscher, Realizing smart scanning transmission electron microscopy using high performance computing, *Rev. Sci. Instrum.*, 2024, **95**(10), 103701.
- 15 A. Ghosh, K. Roccapirore, M. G. Boebinger, D. Mukherjee, A. Al-Najjar, M. McDonnell, S. V. Kalinin and M. Ziatdinov, Integrating high-performance computing with electron microscopy for scientific insights, *Microsc. Microanal.*, 2024, **30**, ozae044.162.
- 16 U. Pratiush, K. M. Roccapirore, Y. Liu, S. V. Kalinin and G. Duscher, Realizing smart stem via machine learning on remote high performance computer, *Microsc. Microanal.*, 2024, **30**, ozae044.164.
- 17 Y. Tsarfati, K. C. Bustillo, B. H. Savitzky, A. Marks, I. McCulloch, C. Ophus, A. M. Minor and A. Salleo, Charge-induced structural rearrangements in organic mixed ionic electronic conductors: a cryogenic 4D-stem study, *Microsc. Microanal.*, 2024, **30**, ozae044.979.
- 18 Y. Tsarfati, K. C. Bustillo, B. H. Savitzky, C. Ophus, I. McCulloch, A. Salleo and A. M. Minor, Structural study of hydrated organic mixed ionic electronic conductors using cryogenic 4D-stem, *Microsc. Microanal.*, 2023, **29**, 264–265.
- 19 Y. Bi, Y. Tian, M. Xu, E. Boltynjuk, L. Velasco Estrada, H. Hahn, J. Han, D. J. Srolovitz and X. Pan, In situ 4D-stem study on grain boundary dynamics in polycrystals, *Microsc. Microanal.*, 2023, **29**, 1509–1510.
- 20 R. Winkler, A. Zintler, O. Recalde-Benitez, T. Jiang, D. Nasiou, E. Adabifiroozjaei, P. Schreyer, T. Kim, E. Piros and N. Kaiser, *et al.*, Texture transfer in dielectric layers via nanocrystalline networks: insights from in situ 4D-stem, *Nano Lett.*, 2024, **24**(10), 2998–3004.
- 21 B. César da Silva, Z. Sadre Momtaz, E. Monroy, H. Okuno, J.-L. Rouviere, D. Cooper and M. Ilse Den Hertog, Assessment of active dopants and p–n junction abruptness using in situ biased 4D-stem, *Nano Lett.*, 2022, **22**(23), 9544–9550.
- 22 H.-Y. Chao, K. Venkatraman, S. Moniri, Y. Jiang, X. Tang, S. Dai, W. Gao, J. Miao and M. Chi, In situ and emerging transmission electron microscopy for catalysis research, *Chem. Rev.*, 2023, **123**(13), 8347–8394.
- 23 Y. Tsarfati, K. C. Bustillo, L. Balhorn, T. J. Quill, J. Donohue, S. E. Zeltmann, B. Savitzky, C. Ophus, C. J. Takacs and I. McCulloch, *et al.*, Microstructural study of organic mixed ionic–electronic conductor thin films using 4D-stem and HRTEM, *Microsc. Microanal.*, 2022, **28**(S1), 50–351.
- 24 I. Arganda-Carreras, V. Kaynig, C. Rueden, K. W. Eliceiri, J. Schindelin, A. Cardona and H. Sebastian Seung, Trainable weka segmentation: a machine learning tool for microscopy pixel classification, *Bioinformatics*, 2017, **33**(15), 2424–2426.
- 25 K. M. Aviles and B. J. Lear, Practical guide to automated tem image analysis for increased accuracy and precision in the measurement of particle size and morphology, *ACS Nanosci. Au*, 2025, **5**, 117–127.
- 26 K. Barakati, H. Yuan, A. Goyal and S. V. Kalinin, Physics-based reward driven image analysis in microscopy, *Digital Discovery*, 2024, **3**(6), 2061–2069.
- 27 S. Hyun Park, S. Park, S. Lee, J. Kim, H. Ahn, B. J. Kim, B. Chae and H. Jung Son, Developement of highly efficient large area organic photovoltaic module: effects of nonfullerene acceptor, *Nano Energy*, 2020, **77**, 105147. ISSN 2211-2855. URL <https://www.sciencedirect.com/science/article/pii/S2211285520307254>.
- 28 Á. Rodrguez-Rodrguez, E. Rebollar, T. A. Ezquerro, M. Castillejo, J. V. Garcia-Ramos and M.-C. Garca-Gutiérrez, Patterning conjugated polymers by laser: Synergy of nanostructure formation in the all-polymer heterojunction P3HT/PCDTBT, *Langmuir*, 2018, **34**(1), 115–125.
- 29 Z. M. Beiley, E. T. Hoke, R. Noriega, J. Dacuña, G. F. Burkhard, J. A. Bartelt, A. Salleo, M. F. Toney and M. D. McGehee, Morphology-dependent trap formation in high performance polymer bulk heterojunction solar cells, *Adv. Energy Mater.*, 2011, **1**(5), 954–962.
- 30 R. Xie, M. P. Aplan, N. J. Caggiano, A. R. Weisen, T. Su, C. Müller, M. Segad, R. H. Colby and E. D. Gomez, Local Chain Alignment via Nematic Ordering Reduces Chain Entanglement in Conjugated Polymers, *Macromolecules*, 2018, **51**(24), 10271–10284.
- 31 R. Xie, Y. Lee, M. P. Aplan, N. J. Caggiano, C. Müller, R. H. Colby and E. D. Gomez, Glass Transition Temperature of Conjugated Polymers by Oscillatory Shear Rheometry, *Macromolecules*, 2017, **50**(13), 5146–5154.
- 32 R. A. Fair, R. Xie, Y. Lee, R. H. Colby and E. D. Gomez, Molecular weight characterization of conjugated polymers through gel permeation chromatography and static light scattering, *ACS Appl. Polym. Mater.*, 2021, **3**(9), 4572–4578.
- 33 S. Van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart and T. Yu, Scikit-image: Image processing in Python, *PeerJ*, 2014, **2**, e453.
- 34 P. I. Frazier, A tutorial on bayesian optimization, *arXiv*, 2018, preprint, arXiv:1807.02811, DOI: 10.48550/arXiv.1807.02811.
- 35 C. Edward Rasmussen, Gaussian processes in machine learning, *Summer school on machine learning*, Springer, 2003, pp. 63–71.
- 36 D. R. Jones, M. Schonlau and W. J. Welch, Efficient global optimization of expensive black-box functions, *J. Global Opt.*, 1998, **13**, 455–492.
- 37 A. Dutta and A. Zisserman, The VIA annotation software for images, audio and video, *Proceedings of the 27th ACM*



*International Conference on Multimedia*, MM'19, ACM, New York, NY, USA, 2019, DOI: [10.1145/3343031.3350535](https://doi.org/10.1145/3343031.3350535), ISBN 978-1-4503-6889-6/19/10.

38 T. Head, M. Kumar, H. Nahrstaedt, G. Louppe and I. Shcherbatyi, *scikit-optimize/scikit-optimize* (v0. 9.0), Zenodo, 2021, Available online: <https://zenodo.org/record/5565057#.Y8Y-iRVBxPY> (accessed on 1 July 2021).

39 X. Lu, H. Hlaing, D. S. Germack, J. Peet, W. Ho Jo, D. Andrienko, K. Kremer and B. M. Ocko, Bilayer order in a polycarbazole-conjugated polymer, *Nat. Commun.*, 2012, **3**(1), 795.

40 R. A. Fair, D. Gamdha, J. T. Del Mundo, A. M. Fenton, A. OConnell, K. C. Bustillo, A. M. Minor, B. Ganapathysubramanian and E. D. Gomez, Automated tem reveals inter-crystalline correlations of conjugated polymers, 2025 Submitted, in review.

41 S. Van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart and T. Yu, scikit-image: image processing in python, *PeerJ*, 2014, **2**, e453.

42 G. Bradski and A. Kaehler, *et al.*, OpenCV, *Dr Dobbs J. Software Tools*, 2000, **3**(2).

43 J. Nunez-Iglesias, A. J. Blanch, O. Looker, M. W. Dixon and L. Tilley., A new python library to analyse skeleton images confirms malaria parasite remodelling of the red blood cell membrane skeleton, *PeerJ*, 2018, **6**, e4312.

44 P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser and J. Bright, *et al.*, Scipy 1.0: fundamental algorithms for scientific computing in python, *Nat. Methods*, 2020, **17**(3), 261–272.

45 C. R. Harris, K. Jarrod Millman, S. J. Van Der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg and N. J. Smith, *et al.*, Array programming with numpy, *Nature*, 2020, **585**(7825), 357–362.

