ACS OMEGA

Open Access

http://pubs.acs.org/journal/acsodf

Article

Benchmarking Unsupervised Clustering Algorithms for Atomic Force Microscopy Data on Polyhydroxyalkanoate Films

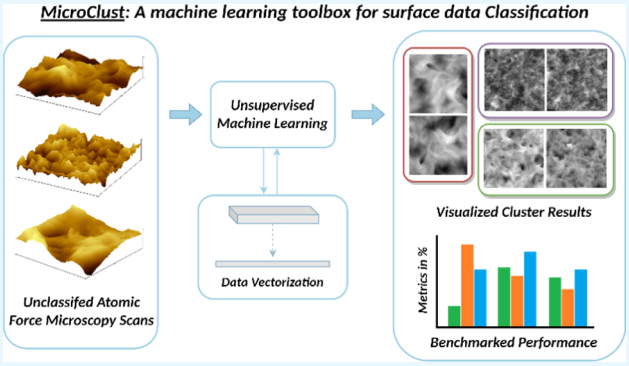
Ashish T. S. Ireddy,* Fares D. E. Ghorabe, Ekaterina I. Shishatskaya, Galina A. Ryltseva, Alexey E. Dudaev, Dmitry A. Kozodaev, Michael Nosonovsky,* Ekaterina V. Skorb, and Pavel S. Zun*

Cite This: *ACS Omega* 2024, 9, 21595−21611

Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

ABSTRACT: Surface of polyhydroxyalkanoate (PHA) films of varying monomer compositions are analyzed using atomic force microscopy (AFM) and unsupervised machine learning (ML) algorithms to investigate and classify films based on global attributes such as the scan size, film thickness, and monomer type. The experiment provides benchmarked results for 12 of the most widely used clustering algorithms via a hybrid investigation approach while highlighting the impact of using the Fourier transform (FT) on high-dimensional vectorized data for classification on various pools of data. Our findings indicate that the use of a one-dimensional (1D) FT of vectorized data produces the most accurate outcome. The experiment also provides insights into case-by-case investigations of algorithm performances and the impact of various data pools. Lastly, we show an early version of our tool aimed at investigating surfaces using ML approaches and discuss the results of our current experiment to configure future improvements.



INTRODUCTION

Polymers have a multitude of uses in the modern era. From prosthetics to drug delivery systems and from general-purpose applications to industrial-grade equipment, they are used in many sectors that serve to improve human life. Polymer compounds are principally differentiated by their chemical and structural composition while commonly being grouped as thermoplastics, elastomers, and thermosets, with each group having its dedicated uses.[1] Thermoplastics can be termed the most widely used group of polymers that have been integrated into daily life. Biron[2] elaborates on the practical benefits and economic aspects of using thermoplastics in a wide range of applications.

The discovery of polyhydroxyalkanoates (PHAs), microbial polymers that are thermoplastic, biodegradable, and biocompatible, marked a significant milestone in the development of novel materials.[3−5] PHAs possess remarkable qualities such as resistance to ultraviolet (UV) radiation, stability in liquid environments, and versatility in processing techniques, including solution, emulsion, powder, and melt methods.[6] PHAs hold the greatest promise for developing biomedical products and devices, including nonwoven and disposable products, sutures and wound dressings, controlled drug delivery systems, scaffolds for tissue engineering, and components for reconstructive surgery and implantation.[7,8] They offer immense potential for regenerating damaged skin, repairing defects in soft tissues, bone engineering, and cardiovascular applications including constructing blood vessels and heart valves.[9]

When aiming to use PHA polymers for distinct applications, researchers need to go beyond the chemical structure to analyze other descriptive characteristics such as physical, mechanical, surface, and electrical properties, etc. Each attribute has a specific relevance to the final application. Poly-3-hydroxybutyrate (P3HB) and its copolymer, poly(3-hydroxybutyrate-*co*-3-hydroxyvalerate) (P3HB-*co*-3HV), are garnering attention in biomaterials for their biocompatibility, with applications in medicine and agriculture for eco-friendly mulches and fertilizers. In manufacturing, they offer alternatives to conventional plastics, supporting sustainability by reducing plastic waste and pollution, especially in packaging and three-dimensional (3D) printing.

P3HB is a homopolymer known for its high biocompatibility, attributed to the natural occurrence of hydroxybutyric acid in cells and tissues of higher animals and humans.[10] However, its utility is limited by its high degree of crystallinity, resulting in rigid products prone to physical aging.[11] To overcome these

Received: March 14, 2024
Revised: April 11, 2024
Accepted: April 12, 2024
Published: April 29, 2024

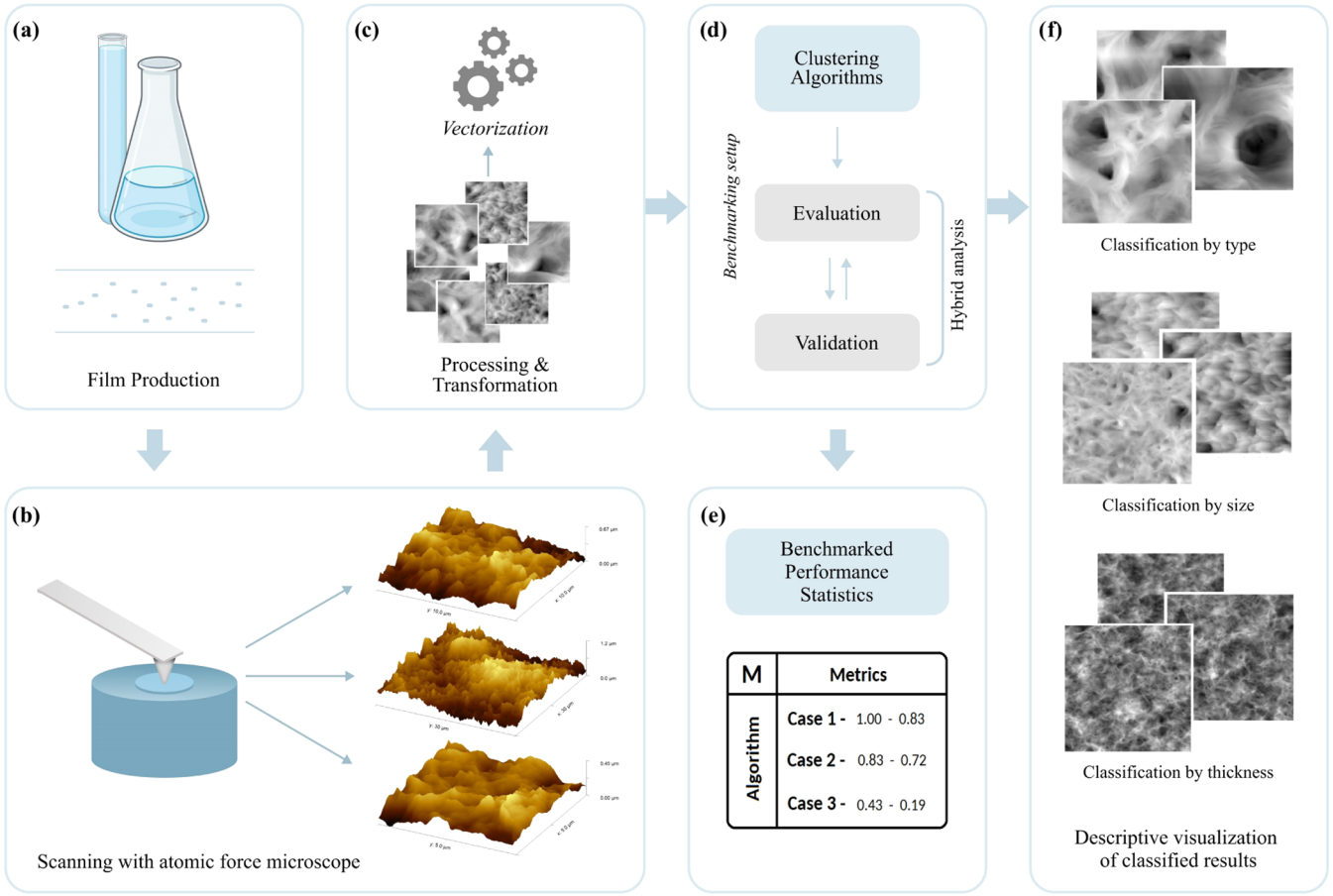


Figure 1. Schematic of the benchmarking experiment: (a) synthesis of PHA films, (b) generation of surface scans using AFM, (c) processing and vectorization of scanned data for algorithms, (d) classification of data using clustering algorithms and validation of results using the hybrid approach, (e) generation of performance statistics and metrics from results generated via the algorithm setup, and (f) visualization and extraction of benchmarking results.

limitations, researchers have explored copolymers such as P3HB-*co*-3HV. This copolymer introduces 3HV monomer units alongside 3HB, offering improved properties such as reduced crystallinity, increased flexibility, and enhanced impact strength without a marked decrease in biocompatibility.[11−13]

Atomic force microscopy (AFM) has played a crucial role as a tool to investigate surfaces at the micro- and nanoscales.[14] The AFM produces high-resolution images of the scanned surface as a collection of height measurements over a selected area. When employed to investigate polymer surfaces, the scanning of surfaces is performed only after the chemical compound is synthesized into films, i.e., the chemical properties of the compounds are usually already known to the researcher by the time of scanning. Nevertheless, when investigating finer details or comparing with other compounds, the amount of time and data demanded by traditional techniques and supervised algorithms is huge. Hence, by employing unsupervised machine learning (ML) algorithms, we can explore and investigate surfaces to obtain standardized results as well as insights that may not be obtained by traditional approaches. In many cases, traditional approaches require meticulous iterations of analysis to identify a unique behavior, while unsupervised approaches can highlight specific behaviors with better efficiency and effectiveness.[15]

AFM has been a crucial tool that has aided researchers in analyzing the surface and morphological properties of polymers without the need to perform in-the-field experiments to understand the details of the polymer's physical properties. AFM allows users to image surfaces down to 10^{-9} m (1 nm) resolution, which provides precise information about the surface texture itself. Using this data, more information can be extracted that can improve understanding of the descriptive properties of the polymer, such as Young's modulus, surface roughness, etc.[16,17] Thus, Zhukov et al.[18] analyzed atomic force microscopy scans of brass samples to study surface properties and have followed up with this approach to analyze the roughness of polyelectrolyte samples using machine learning methods to define possible correlations between surface roughness with the number of layers of polyelectrolytes.[19] A handful of related works describe the usage of ML and deep learning in various scenarios,[20] including several exemplar applications of using image analysis in combination with advanced computing methodologies to generate results. For example, Bolshakova et al.[21] described the basis of using AFM as a tool to explore the properties of bacterial surfaces, while Aldritt et al.[22] used AFM to discover the organic structure of camphor molecules via a deep learning approach. Several works report analysis during and after the scanning process[18,19,21−25] that focus on investigating specific compounds using selective methods. When investigating surfaces, minor changes to the chemical composition can lead to significantly different outcomes of the sample's surface characteristics, therefore altering properties such as friction, adhesion, biocompatibility, etc.[26−29]

https://doi.org/10.1021/acsomega.4c02502
ACS Omega 2024, 9, 21595−21611

In the present work, we will compare the results of using unsupervised ML algorithms to characterize multicomponent PHA films using data acquired from AFM because the biocompatibility of this class of biopolymers depends on certain monomer inclusions and surface properties. We will classify images of scanned surfaces for their properties, such as the scan area, film thickness, and monomer type. We will compare the benchmarked performance of clustering algorithms when surface data are used for classification and evaluate the results using our own hybrid approach. The paper is divided as follows: Experimental Section consists of the methodology and implementation of our benchmarking setup. Results Section dives into the obtained results after classification. Discussion Section is the conclusion and outline of future work.

EXPERIMENTAL SECTION

The workflow of this study is outlined in Figure 1. This section elaborates on the procedure of synthesizing PHA films and the setup to acquire surface scans via atomic force microscopy. It is followed by a description of postprocessing operations to prepare the data for benchmarking.

Polymer Film Production and AFM Operation. We used PHA films in our experiment. PHA is a biodegradable polymer derived from *Cupriavidus necator B-10646 strain.*[30] This microorganism was employed for synthesizing PHA polymers in high yields, through which the resulting composition of homopolymers, namely, P3HB [100%] and copolymers poly-3-hydroxybutyrate and poly-3-hydroxyvalearte (P3HB-*co*-3HV [90:10%]), was obtained. The culture medium and cultivation conditions for synthesizing PHA copolymers were meticulously controlled. Glucose served as the main carbon source in the Schlegel medium,[34] supplemented with precursors such as 1,4-butanediol and salts of valeric acids. These precursor substrates were added incrementally to mitigate their toxic effects on the cell culture. The pH of the medium was maintained between 7.0 and 7.2, and cells were grown in the batch culture mode. Inoculum preparation involved resuspending the stock culture grown in mineral solution, with glucose concentrations ranging from 5 to 10 g/L.[31] Cell growth occurred in two distinct phases: Phase 1 involved growth in the Schlegel medium with a limited nitrogen supply, followed by Phase 2 in a nitrogen-free medium for the same duration to activate the polymer accumulation process. Cultivation parameters were optimized based on the physiological effects of precursor substrates and their concentrations on cell growth and PHA yield. The bacterial culture synthesis of PHA copolymers operated as a multifactorial system, with excess carbon source, limited nitrogen supply, and controlled concentrations of toxic precursor substrates.[32]

PHA recovery from the cell biomass involved a two-stage process. First, lipids and fatty acids were removed by using ethanol, followed by polymer extraction with dichloromethane. The extracted polymer was then precipitated with hexane, and its content was determined by using gas chromatography. Further purification steps included redissolving the polymer in chloroform and precipitating it with isopropanol or hexane before drying at 40 °C.

The chemical composition of PHA was determined by chromatography of the methyl esters of fatty acids after the methanolysis of cell biomass. Similarly, the purity and composition of the polymer were analyzed by chromatography of methyl esters of fatty acids after the methanolysis of purified polymer samples. Methanolysis involved boiling the polymer sample with chloroform, methanol, and concentrated sulfuric acid under reflux condensers for 160 min, followed by analysis of the chloroform layer.[33]

The set of solutions was prepared and filtered and then poured onto Petri dishes ensuring consistent distribution, after which they were left undisturbed in controlled settings for solvent evaporation, therefore creating films of varying thicknesses corresponding to the initial polymer concentrations. The thicknesses of each film are measured using a digital micrometer (*Legioner EDM-25-0.001*) and their respective measurements are presented in Table S1.

Imaging of the PHA films was performed using the NT-MDT AFM. The scanning of films was conducted in semicontact mode, which allows for high-resolution imaging while minimizing potential damage to the sample surface. The instrument was set to a high-signal-height mode to ensure accurate measurements and operated at a resolution of 512 pixels per image side to capture intricate surface details. Before scanning, the AFM probe was calibrated to guarantee precise and reliable results, and the scanning process was carried out at a controlled room temperature to mitigate environmental effects on the measurements. The raw scanned data was processed using the Gwyddion toolkit,[35] where four correction operations were carried out to enhance the overall quality and eliminate possible errors on the surface. The operations were performed on all scanned samples and are as following.

1. ***Plane leveling:*** This operation performs uniform leveling of the scanned image by subtracting a collectively calculated plane from the image. It is aimed at fixing the degree of curvature that occurs during the scanning process.
2. ***Face leveling:*** An enhanced version of plane leveling that performs uniform leveling of the surface to obtain the best possible horizontal plane when there are large objects present on the surface. The main difference between plane leveling and face leveling is the sensitivity to height differences. Face leveling is not suitable when applied to images that have noise, random artifacts, or height differences of many magnitude orders within the same space.
3. ***Alignment of Rows:*** The operation minimizes line differences (i.e., remaining horizontal lines after plane correction) based on a function that gives more weight to flat areas than areas with large slopes, thereby making the surface uniform without horizontal lines.
4. ***Removal of Scars:*** This operation fixes line defects occurring due to scanning and scratching from the cantilever. The scratches are removed by compensating for the gaps with neighboring lines, thereby providing a clear and uniform surface.

Upon the completion of preprocessing, the data was exported to the dimensions of 512 × 512 where each point corresponds to the height data of the sample. The next stage involves the transformation of the data into a vector form and performing one-dimensional and two-dimensional Fourier transformations specific to each case of clustering.

Implementation. We studied 52 AFM scan samples of both P3HB and P3HB-*co*-3HV polymers, respectively (a total of 104 scan samples). These polymers were scanned at a resolution of 512 × 512 pixels (length × width) in scan dimensions of 5 μm × 5 μm, 10 μm × 10 μm, and 30 μm × 30 μm. After processing with the Gwyddion toolkit, the scans were exported in.txt format and further used within our benchmarking experiment for

https://doi.org/10.1021/acsomega.4c02502
ACS Omega 2024, 9, 21595−21611

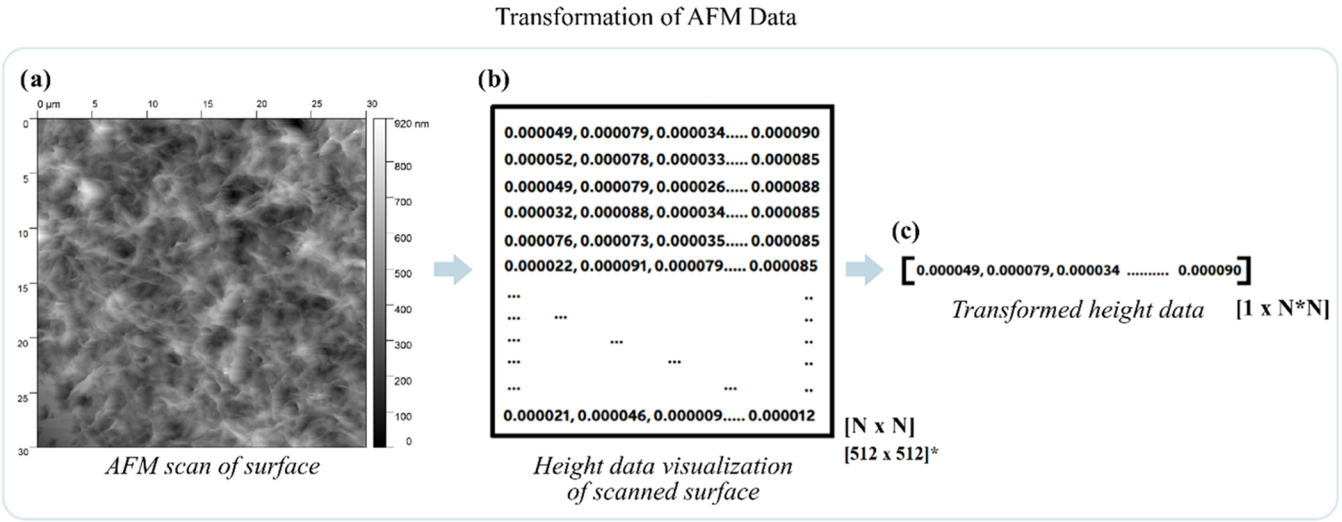


Figure 2. Schematic describing transformation of AFM scans into vectors for classification (i.e., Vectorized Data (VD)): (a) processed AFM scan surface, (b) numeric representation of the AFM scan describing each height point, and (c) transformed height data to be fed to the clustering algorithms.

simulations. Figure 2 shows the basic flow of the data vectorization. To validate and verify the results of our experiments, we have implemented a hybrid analysis approach that utilizes prior knowledge of the scans and their respective properties (i.e., type of polymer, size of scan, etc.) as a reference to validate the generated results. From a total of 104 AFM scans, our data is factually classified based on the following.

- Two polymers (P3HB and P3HB-CO-3HV)—implying two clusters.
- Three sizes of scans (5, 10, and 30 μm)—implying three clusters.
- Six thicknesses of films—implying six clusters.

Our toolkit MicroClust[1] is aimed at aiding researchers in analyzing small-scale surfaces by extracting and investigating possible relations existing within the data using machine learning algorithms. In the current release of our toolkit, we have created a criterion that requires all preprocessed AFM scan data to have homogeneous dimensions (i.e., length and width must be the same for all input samples). Upon fulfillment of this condition, the transformation and clustering operations commence. Transformation of the processed data is the first operation performed within the toolkit, where the data is restructured into a single vector such that it can be accepted into the clustering algorithms as single entities (i.e., vector form rather than 2D data types). In future versions of our toolkit, we plan to extend and modify this approach by allowing resizing of scans and usage of heterogeneous scan data for classification. There are three transformations of data that we have performed:

1. Vectorized data (VD),
2. One-dimensional discrete fast Fourier transform of Vectorized Data (1D-FFT),
3. Two-dimensional discrete fast Fourier transform of Vectorized Data (2D-FFT).

The data acquired after processing via Gwyddion is reformatted into a vector array by concatenating each row of scan data consecutively after the other in the dimension of the scan size (i.e., for scan data with dimensions 512 × 512, the resulting vector will be the product of the length of the rows and columns; therefore, 262,144 is the length of the vector). Within our work, the transformed data without any modifications are termed Vectorized Data (VD). For the one-dimensional fast Fourier transform (1D-FFT), we extract height values from the Vectorized Data (VD) in the orientations of rows and columns individually. Fourier transform and logarithmic conversion are applied individually for either vector of data. The two-dimensional fast Fourier transform (2D-FFT) follows a different approach in which the transform of VD is directly applied to the source scan data. After this, the transformed data is extracted in the same format as VD and 1D-FFT. We have addressed the rationale of using the Fourier transform and the observations of using data in the orientation of rows and columns in the Discussion and Results Sections, respectively. The data acquired after transformations are fed as input for the clustering algorithms.

We chose Python as the primary platform to implement our toolkit and perform benchmarking of clustering algorithms. The clustering environment used in our tool is constructed using the Scikit-learn library.[36] We have implemented 12 of the most widely used clustering algorithms, and we use common metrics that match our application.[37,38] Despite the unsupervised learning nature of the algorithms, a handful of them still require partial tuning to achieve relevant results as per the application (i.e., they require parameters and boundary conditions to be provided by the user to produce meaningful results, e.g., minimum distance between clusters and centroid distances). As this work aims to showcase the benchmarked performance statistics of the algorithms, we have explicitly tuned the clustering algorithms to obtain a maximally broad insight without redundancies in its design approaches. However, in future versions of our tool, we aim to produce clustering results without the need for user input by introducing investigative architectures that explore the input data and metrics to find feasible explanations for the results. Following are the algorithms that have been used in this experiment and descriptions of their working principle.

1. ***K-means*** algorithm[39] clusters elements based on calculating the distance from each data point to the centroid and assigning it to a cluster based on the respective distance. It is one of the most widely used algorithms in the field for various applications. It is crucial to observe its perform-

https://doi.org/10.1021/acsomega.4c02502
ACS Omega 2024, 9, 21595−21611

ance in situations where the data points might not have convex cluster shapes. K-means algorithm is differentiated by its approach to clustering where cluster centers in the algorithm are randomly assigned and then converge to the local minima, thereby generating the results: this makes the final results dependent on the initialization of the centroids.

2. ***K-means***++ algorithm[40] works on the same principle of K-means and is an extension where it improves the selection of centroids by using a calculated approach for the selection of cluster centers. It follows the greedy approach by selecting the initial cluster centroids based on the probability distribution of the points that contribute to the overall inertia. Hence, this algorithm takes more trials to identify the best centroid over random selection unlike in K-means.
3. ***Bisect K-means*** algorithm[41] is another variant of K-means that works by selecting centroids based on progressive steps resulting from previously identified clusters. It uses a top-down approach where a huge cluster is split into smaller clusters and iterates until the target number of clusters is reached. We performed our experiments using the "*biggest inertia*" method that creates clusters based on similar cluster sizes.
4. ***Hierarchy*** algorithm[36] works by grouping elements using a linkage criterion. The linkage criterion is a method to measure the distance between elements of a cluster (within the cluster and between clusters): this method is also known as agglomerative clustering. We have used the bottom-up approach to obtain clustering results, where initially each element is considered unique after which they are merged based on the linkage criteria. The results of this algorithm are represented using a dendrogram. We have implemented this algorithm using four types of linkage metrics as below.
    - *Ward*: Merges elements into a single cluster based on the minimized sum of squared differences within all clusters.
    - *Maximum linkage*: Merges elements based on the maximum distance between pairs of clusters.
    - *Average linkage*: Merges elements based on the average distance between all observations.
    - *Single linkage*: Merges elements based on the closest observations between pairs of clusters.

    During our simulations, we selected the ward method of linkage as the high dimensionality of our data produced ambiguous results for other methods.
5. ***Fuzzy C-means***[42] algorithm works by grouping elements based on similarity. It is distinctive as it allows for elements to belong to more than one cluster, which are then grouped over iterations. The current implementation uses the least-squares function to calculate the distance between each element and the cluster center to find the ideal grouping for the elements.
6. ***Spectral*** algorithm[43] functions by analyzing the mutual similarities of the samples by generating eigenvectors from the affinity matrix of the samples. It reduces the data into lower dimensions to make sure that weak variance points (present in high-dimensional data and images) do not have a significant impact on the final clustering results.
7. ***DBSCAN (Density-Based Spatial Clustering of Applications with Noise)*** algorithm[44] works based on the concept of partitioning areas of high-density elements apart from areas of low element density. The algorithm allows for the analysis and creation of clusters that can take up shapes other than convex. The algorithm requires two parameters: **minimum number of samples** (*minimum number of samples in the neighborhood to be considered a cluster center*) and ***epsilon*** (maximum distance between two samples to be considered within the neighborhood of the other). When using this algorithm, there are exceptional situations when parameters are unbalanced that return clustering results that consist of either a single cluster with all elements within or all elements designated as noise. Hence, we have implemented two versions of the DBSCAN algorithm that allow us to analyze a partially automated clustering approach and a completely manual approach.
    - *Automated* where the epsilon value is determined by finding the distance between elements using the K nearest-neighbors, followed by extracting the knee point of the data (elbow method) using the sorted distances, while the minimum number of samples is provided by the user, we followed the implementation by Rahman and Sitanggang[45]
    - *Manual* where the epsilon value and the minimum number of samples are both provided by the user.
8. ***HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise)*** algorithm[46] extends DBSCAN and finds an optimal distribution when clusters of varying densities exist. The algorithm requires us to provide the ***minimum cluster size***, the ***minimum number of samples for core point existence***, and ***epsilon***—a distance threshold below which clusters will be merged into a single unit. This approach works based on generating a mutual reachability matrix by using the core distance of a sample and the distance to *n*th nearest sample, where smaller distances lesser than the threshold are treated as noise.
9. ***Mean Shift*** algorithm[47] follows the K-means algorithm technique of centroid-based clustering, i.e., defining cluster elements based on the central points. The algorithm works by selecting the centroids for each given region based on the mean of elements present within the same region. The ideal centroid is found by iteratively varying the possible positions by finding the local maximum of the estimated probability density. The parameter to be fixed here is the **bandwidth**, which controls the shift of the area of coverage into a higher-density region until convergence and determination of the cluster centers.
10. ***OPTICS (Ordering Points To Identify Clustering Structure)*** algorithm[48] works similarly to DBSCAN where high-density core samples are found and newer members are added. It requires us to provide the ***minimum number of samples to be considered a core point of the cluster*** and the ***maximum epsilon*** (i.e., the maximum distance between two samples to be considered in the neighborhood of the other). Here, the addition of new elements into a cluster is based on the calculated reachability distance and a ranking spot based on the same distance. The main difference between OPTICS and DBSCAN is the usage of reachability distances, therefore allowing more elements to be considered within the cluster than noise.

https://doi.org/10.1021/acsomega.4c02502
ACS Omega 2024, 9, 21595−21611

Table 1. Clustering Results Analysis Metrics and Their Derived Significance

#	Type	Name of metric	Definition/significance	Bounds	Ideally expected result
1.	Ground truth-based metrics (M1)	Rand index	Measures similarity using labels	[0.0, 1.0]	1.0 = perfect match
2.		adjusted Rand index	measures similarity using labels (considering permutations)	[−0.5, 1.0]	−0.5 to 0.0 = poor prediction 1.0 = perfect prediction
3.		adjusted mutual information score	measures similarity using labels (considering a balanced distribution of clusters)	[−∞, 1.0]	1.0 = perfect prediction [−∞, 0.0] = random labeling/no agreement
4.		homogeneity	measures if all clusters have elements belonging to a single class	[0.0, 1.0]	1.0 = homogeneous prediction
5.		completeness	measures if all elements of a class belong to the same cluster	[0.0, 1.0]	1.0 = complete prediction
6.		V-measure	harmonic mean of homogeneity and completeness	[0.0, 1.0]	1.0 = homogeneous and complete prediction
7.		Fowlkes Mallows score	measures similarity of clusters using geometric mean of precision and recall	[0.0, 1.0]	0.0 = random labeling; 1.0 = perfect similarity between clusters
8.	self-evaluation metrics (M2)	Silhouette coefficient	evaluates the structure of clustering	[−1.0, 1.0]	−1 = incorrect prediction 0 = overlapping clusters 1 = best prediction
9.		Calinski−Harabasz index	evaluates the ratio of cluster variance	[0, ∞]	higher values indicate well-separated clusters
10.		Davies Bouldin index	evaluates the similarity/separation of the clusters	[0.0, ∞]	0.0 = best clustering

11 ***Affinity Propagation*** algorithm[49] approaches clustering from the perspective of measuring the distance between mutual samples. An exemplar set of samples is selected, and a message (ping) is sent toward all other samples. Based on the response obtained, a consensus is created in the form of a similarity matrix that selects the ideal point that has the best response of ping from all elements to be clustered. The parameter to be provided here is the ***damping factor*** that varies the sensitivity of the receiving point.

12 ***BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies)*** algorithm[50] works on the basis of the data reduction method. It creates a feature tree with nodes and leaves, where each element is initially a leaf, and the cluster centroids are selected from one of the leaves. This is followed by iteratively constructing the whole tree. Optionally, there are two parameters that can be varied, cluster merger threshold and the branching factor that sets the maximum number of nodes after which the node is split into two. Our current implementation had fixed iterations of 150, a fuzzy partition component of 2.0, and a tolerance criterion of 10^{-5}.

Within our implementation of the toolbox, we have fixed the random seed state for all algorithms to a specific value to ensure that consistent results are obtained on every run. We confirmed this by performing a single test run of clustering polymers by size using 21 different random seed values and calculating the standard deviation for all trials. The calculated standard deviation was in the range of 0.59−3.01%; hence, a fixed value of random seeds was selected.

The algorithms in this benchmarking experiment require a minimum of one criterion (i.e., the number of clusters demanded) and up to three hyperparameters for a few of them. To obtain maximal insight from our benchmarking experiment, we split the clustering operations into two formats. This is done to verify whether the algorithms can distinguish the data as per our existing conditions (i.e., differentiable by polymer type, size of scan, or film thickness) and study other resulting behavior. The two formats are as follows.

1. ***One-step Clustering (F1)*:** In this format of clustering, a single iteration of grouping is performed with exclusive cluster thresholds of either 2, 3, or 6, demanded from the algorithms (i.e., the algorithms are required to group the elements into the specified clusters based on the properties of the data given).
2. ***Iterative Clustering (F2)***: Also termed as divisive clustering, this format classifies the data in iterative cycles based on the results generated at each wave (i.e., a group of elements is successively clustered into superclusters until a saturation point is reached). In our case, the saturation point was defined as the stage when the number of required clusters is greater than the number of data samples remaining for classification.

Upon completion, the algorithms return a vector of labels with numbers $[0, 1, 2, \cdots, N-1]$, where N is the desired number of clusters, denoting which cluster each element belongs to. To evaluate the accuracy of the results, the generated labels and the source data are fed to the algorithms separated into two groups: *implicitly* and *explicitly* tuned algorithms corresponding to the input required for classification. Implicitly tuned algorithms are those that require only the number of clusters to be provided as input, while explicitly tuned algorithms require more than one parameter to be optimized. Further, the results from the algorithms are validated using two types of metrics as below.

1. **Ground truth-based metrics (M1).** These metrics *require prior knowledge about the data* to verify the results generated by the algorithms. We have used this category of metrics to verify the results obtained and as a way to measure the performance of the algorithms. However, in real-world practical applications, the ground truth information may be unavailable. Unsupervised clustering is often implemented to discover the possible ground truth; hence, the usage of these metrics is specific to investigating the benchmarking results and may be considered as an optional feature for the toolkit.
2. **Self-evaluation metrics (M2).** These are metrics that evaluate the result using the data provided as input and

https://doi.org/10.1021/acsomega.4c02502
ACS Omega 2024, 9, 21595−21611

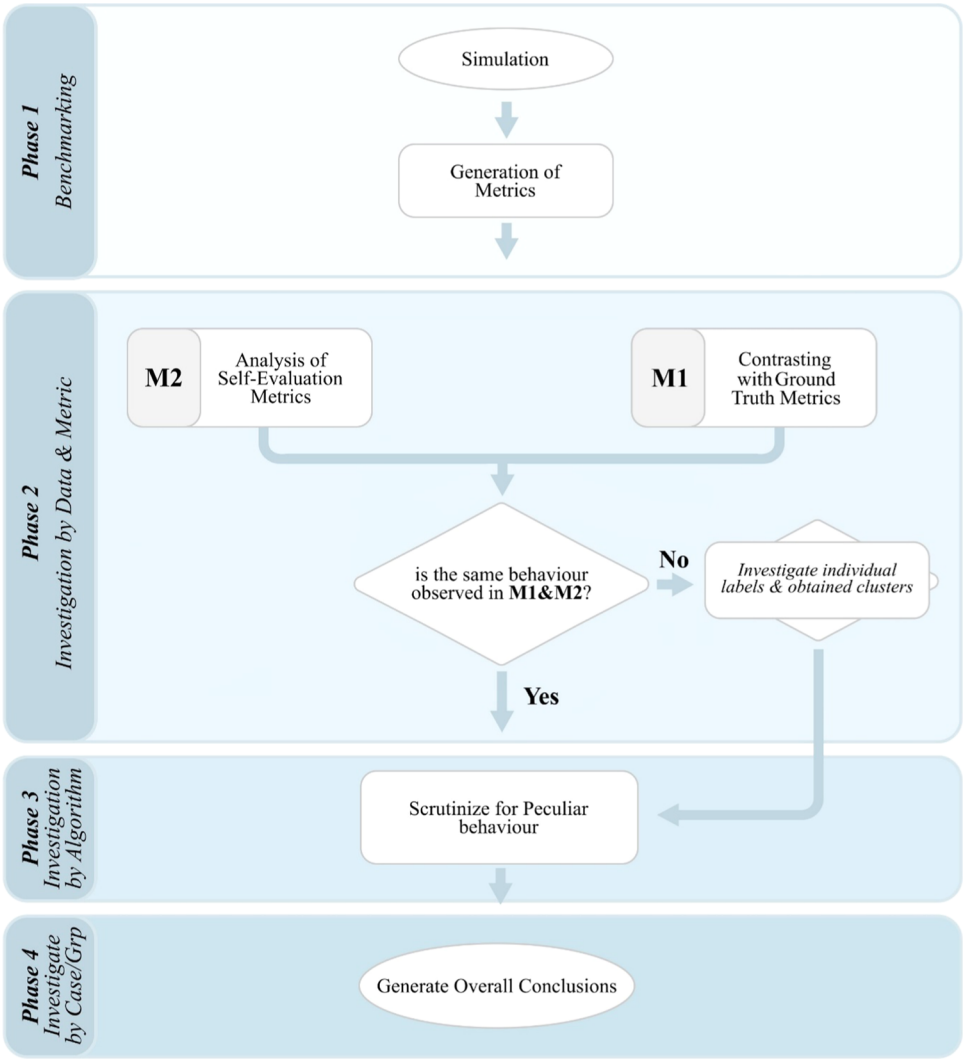


Figure 3. Schematic flowchart describing the hybrid approach to evaluate clustering results using self-evaluation metrics M2 and ground truth data M1.

the clustering model itself, without the requirement of ground truth knowledge.

There are many metrics available in the literature that describe the internal and external qualities of clustering via mathematically derived solutions[51] (e.g., scatter criteria, Pearson correlation measure). As our aim is to create an open-source toolkit, the results from our first iteration were resolved to draw conclusions via metrics; hence, we have selected broad metrics that can be directly correlated with rational definitions such as similarity, variance, completeness of clusters, etc. In future versions of our tool, we aim to address and implement other metrics for analysis. Table 1 provides a simplified understanding of the implemented metrics, their significance, bounds of scores, and the interpretation of the expected results to measure performance.

Rand Index (RI), Adjusted Rand Index (ARI), and Adjusted Mutual Information Score (AMI).[52,53] These are metrics that analyze and measure the similarity of the expected results versus the generated results. Where the Rand Index directly measures the similarity using the predicted data and the expected data, the Adjusted Rand Index measures similarity by considering the possibility that the clustering might have occurred due to chance. The Mutual Information score analyzes the result using the distribution of clusters and elements within.

Homogeneity (H), Completeness (C), and V-Measure (V-M).[54] These are metrics that analyze and measure the similarity of the clustering results based on conditional entropies of the generated results against expected results. The homogeneity measure is used to analyze if clusters exclusively contain data points that are members of a single class, and the Completeness measures if the data points of a given class are elements belonging to the same cluster. The V-measure is the harmonic mean generated from the Homogeneity and Completeness metrics.

Fowlkes Mallows Score is the Geometric Mean of Precision and Recall (GM). This is used to measure the similarity of clusters obtained by calculating the TP, FP, TN, and FN values as per the labels provided and generated. Unlike other metrics, this approach does not consider the cluster structure during computation and is a viable option when comparing clustering algorithms such as k-means, which assumes isotropic blob shapes with algorithms that produce results in various folded shapes such as the spectral algorithm.

Silhouette Score (SS).[36] It is a metric that evaluates the structure of clustering (i.e., assigns a score based on the appropriate clustering concentration, higher scores for well-defined clusters). Dense clustering is indicated by +1, incorrect clustering is indicated by −1, and values close to 0 indicate

https://doi.org/10.1021/acsomega.4c02502
ACS Omega 2024, 9, 21595−21611

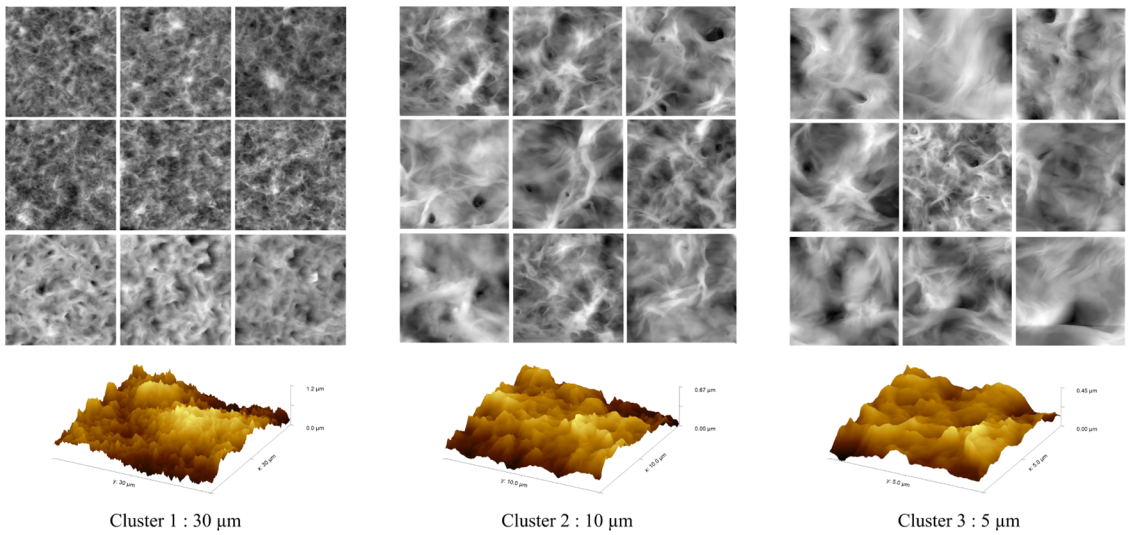


Figure 4. Visualized results of K-means++ algorithm when clustering P3BH and P3HB-CO-P3HV data according to their scan size. When visualized, there are notable features that differentiate 30 μm scans against the 10 and 5 μm scans.

overlapping clusters. Well-separated and dense clusters give better scores.

Calinski–Harabasz Index (CHI).[36] It measures the variance of the clusters to check if they are well-defined and well-separated. It analyzes the ratio of (sum of intercluster dispersion) divided by the (sum of intracluster dispersion). A good score indicates well-defined clusters that are well-separated, and a higher score is better.

Davies Bouldin Index (DBI).[36] It measures the separation between the clusters. It indicates the average similarity between clusters by measuring the distance between clusters against the size of the respective clusters. The score is calculated based on quantities and features.

Using the AFM data, algorithms, and metrics, we benchmarked the transformed data in F1 and F2 approaches and evaluated them using M1 and M2 metrics to validate the generated results. To investigate the results in greater detail, we used our own hybrid approach to analyze the results and further explore results using different tuning conditions.

Evaluation Metrics. This section elaborates the methods of benchmarking the clustering algorithms and their evaluations based on the metrics described above. The flowchart depicted in Figure 3 explains our approach at analyzing the results obtained from the clustering algorithms. We split the analysis of results into four phases as below.

Phase 1. The results obtained from the clustering algorithms are formatted into an array of labels for analysis using both M1 and M2 metrics. Further, the ground truth of the present data for classification is established and fed into the M1 metrics.

Phase 2. At this stage, the results obtained from the algorithms are analyzed using the metrics of M2 where the expected result is considered as a criterion and compared against the generated results individually per metric.

Followed by individual analysis of M1 metrics. The M1 group of metrics is used to validate and cross-verify the results generated from the M2 metrics and the algorithms themselves. Despite each algorithm generating metrics M1 and M2, we have performed an independent investigation in **Phase 3** in order to make sure that we do not overlook the performance of individual algorithms. Upon generation of M1 and M2 groups of metrics, we aim to check if both groups of metrics follow the same trend, i.e., if the scores match the behavior indicated in M1 as in M2. If the condition is true, the analysis moves into **Phase 3**. Otherwise, a deeper analysis is conducted to investigate instances in which the results are more varied than usual.

Phase 3. At this phase, individual algorithms under each type of input data are scrutinized to find algorithms with good, poor, and consistent performance.

Phase 4. The final phase dives into analyzing the obtained results of M1 and M2 metrics across cases (i.e., comparing the results of using VD, 1D-FFT, 2D-FFT, several thresholds). The cases here also depict the type of information used (i.e., the results of using only P3HB data vs using both P3HB and P3HB-*co*-3HV data together).

RESULTS

To obtain the best results with broad outreach, we have divided the simulations into three cases (signifying the properties of scan size, monomer type, and film thickness). In certain cases, we have subdivided the data into smaller pools to understand the behavior of algorithms and its results when using categorical data. The simulations of our experiment can be found at Github.[2]

Case 1: Classification by Size (Threshold = 3). The expected results in this case are three clusters consisting of AFM scans classified by their size of scanned area (i.e., 5, 10, and 30 μm). Figure 4 shows the results of the K-means++ algorithm when clustering a combined pool of data. As clustering multidimensional data has its own complexities (curse of dimensionality), we created three pools of data to maximize our insight. Pools A and B consist of exclusively P3HB and P3HB-*co*-3HV data, respectively, while pool C is a combination of the two. Initially, we tuned the algorithms to a uniform set of hyperparameters; however, the results generated with such a setup were not accurate or meaningful (i.e., the results did not produce the required number of clusters or did not converge); hence, we recalibrated the hyperparameters for each data pool such that the desired number of clusters can be obtained. To ease notations, we have designated algorithms 1−6 and 7−12 as implicitly and explicitly tuned with reference to the implementation section, where implicitly tuned algorithms require the total number of clusters to be provided as input,

https://doi.org/10.1021/acsomega.4c02502
ACS Omega 2024, 9, 21595−21611

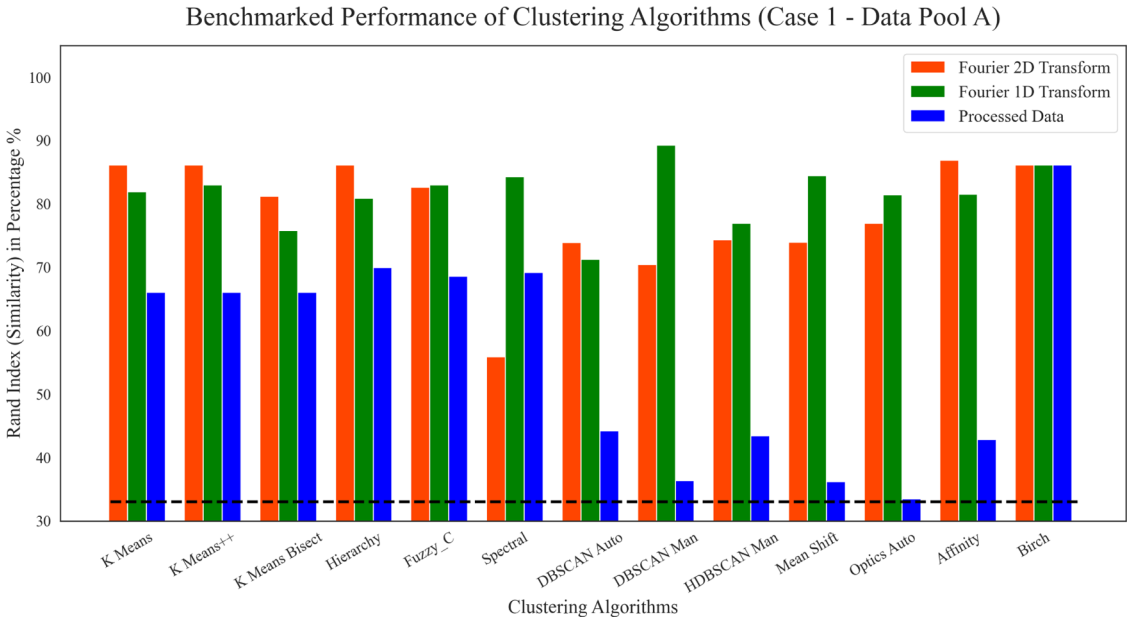


Figure 5. Graphical representation of prediction similarity (Rand Index) for a data pool consisting exclusively of P3HB data when classified by the scan size (Case 1-A). The dotted line at 33.3% denotes the similarity index of random (chance) clustering.

whereas explicitly tuned algorithms require more than 1 hyperparameter to be tuned. On analyzing M2 metrics for all three pools of data, we found that 1D-FFT data had the most well-separated clusters with the least intercluster overlapping.

We contrasted the results of self-evaluation metrics with that of ground truth-based metrics and observed the same behavior where a higher similarity score could be correlated with that of the cluster structure and similarity extracted via the self-evaluation metrics. Across all three data pools, we found 1D-FFT to have the highest average accuracy for all algorithms at 75.29% followed by 71.2% of 2D-FFT, and VD had the lowest accuracy at 54.3%: this can be observed in Figures 5, S1, and S2. On analyzing individually, we also found that the magnitude of self-evaluation metrics (M2) was higher for pool C over the remaining: this can be attributed to the difference in the magnitude of the scanned samples (i.e., the height of scans). Regardless, the behavior was verified by M1 metrics. Table 2 shows the benchmarked performance metrics of data pool C, whereas the benchmarked performance of data pools A and B are shown in Tables S2 and S3.

Upon scrutinizing individual algorithms, we observed that explicitly tuned algorithms were more prone to have errors such as nonconvergence or bulk outliers. The DBSCAN automated algorithm had the lowest score and is a factor of many samples being considered as noise despite lowering the threshold to the smallest level possible. Similarly, the OPTICS algorithm had instances where the data were sparse and the algorithm could not converge despite having the lowest logical threshold set 5. Within our analysis, we also observed that most of the errors in the classification occurred when differentiating data of 5 and 10 μm sizes: this could be verified by the silhouette score when nearing zero, indicating overlapping clusters. Implicitly tuned algorithms had consistent performance in contrast to those that required tuning.

Case 2: Classification by Polymer Type (Threshold = 2). The expected results in this case are two clusters relative to the type of polymers, P3HB and P3HB-CO-3HV. In case 1, we observed the behavior of the algorithms when data are separated by the polymer type; in this case, we created data pools consisting of both polymers distinguished by their scan area. Pool A consists of all scan sizes (5, 10and 30 μm), while pool B consists of exclusively 30 μm scan data. In this case, the probability of a sample being classified versus assigned by chance is at 50% as there are only two possible clusters present.

In either data pool, the self-evaluation metrics indicate the clusters to be separated with a moderate amount of variance between each other, and this is shown in Tables 3 and S4. However, the accuracy produced from data pool A across all three data transformations is lower than that of pool B. We observed an increase in the accuracy of 14.61% for processed data (VD), 16.63% for 1D-FFT, and 5.98% for 2D-FFT. Pool B produced a significantly more accurate result than Pool A, and this is shown in Figure 6, where despite using films of a single scan size, there exist errors within the clusters. On scrutinizing individual algorithms and data transforms, we found that overlapping of results was less likely to occur when using VD over 1D-FFT and 2D-FFT in this case, we assume this may be due to the difference of scaling occurring after Fourier transformation. To test this, we increased the required number of clusters to investigate the results statistically and visually. On demanding up to five clusters using VD, the resulting clusters had less variance of elements over 1D-FFT and 2D-FFT data (i.e., the data is grouped into smaller chunks of clusters rather than being divided individually). This provided the final metric results with clusters that are different from each other. This was also observed visually, where data from within either polymer were subdivided within.

We also paid attention to clustering by chance, as there are only two states for final prediction. There are instances when algorithms group the majority of elements into a single cluster while leaving very few elements as the rest; on one hand, this guarantees a good score for one type of polymer but completely ignores the other (i.e., algorithms classify the majority of elements into a single cluster and label the rest as noise, thereby achieving around 50% of accuracy while the rest is deemed not fit). This is accurately depicted in Figures 7 and S2 with the

https://doi.org/10.1021/acsomega.4c02502
ACS Omega 2024, 9, 21595−21611

Table 2. Benchmarked Performance Metrics of Clustering Algorithms When Classifying a Combined Data Pool of P3HB and P3HB-CO-3HV Data by Scan Size (Case 1—Data Pool C)

	metrics	Rand index (in %)			completeness (in %)			Silhouette score			Calinski–Harabasz index			Davies Bouldin index		
tuning	data transform	VD	1D	2D	VD	1D	2D	VD	1D	2D	VD	1D	2D	VD	1D	2D
	algorithms															
implicit	K-means	52.82	77.00	80.35	13.26	53.70	56.53	0.45	0.42	−0.08	134.07	100.51	16.48	0.90	0.93	2.65
	K-means++	52.62	77.00	80.35	12.57	53.70	56.53	0.46	0.42	−0.08	134.52	100.51	16.48	0.87	0.93	2.65
	K-means BISECT	52.77	63.28	75.93	14.68	25.80	52.72	0.39	0.35	−0.10	113.17	84.11	13.60	1.08	1.13	2.98
	hierarchy	46.89	77.31	79.23	14.22	56.48	61.32	0.51	0.38	−0.09	117.02	91.70	17.98	0.89	0.96	2.03
	Fuzzy C-means	54.71	77.71	79.80	14.18	55.66	52.68	0.42	0.40	−0.06	132.18	98.90	16.76	0.92	0.96	2.67
	spectral	56.87	71.94	55.15	10.52	69.45	0.26	0.07	0.31	−0.06	70.61	46.78	1.44	3.79	0.83	10.67
explicit	DBSCAN (automated)	37.53	76.98	35.33	18.43	77.31	13.03	0.58	0.31	−0.24	42.20	24.99	1.26	1.17	2.60	2.09
	DBSCAN (manual)	37.51	77.64	42.91	18.64	81.95	13.80	0.55	0.31	−0.10	29.10	24.81	7.24	0.94	3.17	1.74
	HDBSCAN	37.56	71.30	64.89	20.01	54.14	32.44	0.55	0.29	−0.18	42.82	32.09	4.36	1.16	1.50	6.87
	mean shift	43.32	74.16	71.50	16.76	63.32	52.39	0.50	0.36	−0.19	81.46	54.69	9.60	1.00	0.89	2.39
	OPTICS	53.68	77.29	33.35	9.28	53.11	19.62	0.32	0.23	0.04	67.52	40.68	0.55	0.98	1.24	1.11
	affinity	41.45	33.35	80.71	20.71	19.62	58.33	0.49	−0.16	−0.07	20.43	0.43	17.32	1.12	1.37	2.47
	BIRCH	79.23	79.23	79.23	61.32	61.32	61.32	−0.09	0.24	−0.09	17.98	51.25	17.98	2.03	1.29	2.03

Table 3. Benchmarked Performance Metrics of Clustering Algorithms When Classifying a Combined Data Pool of P3HB and P3HB-*co*-3HV of All Scan Sizes (5, 10, and 30 μm) as Per Their Type (Case 2—Pool A)

	metrics	Rand index (in %)			Completeness (in %)			Silhoutte score			Calinski–Harabasz index			Davies Bouldin index		
tuning	data transform	VD	1D	2D	VD	1D	2D	VD	1D	2D	VD	1D	2D	VD	1D	2D
	algorithms															
implicit	K-means	57.99	53.37	50.07	33.27	5.91	0.80	0.58	0.43	0.16	152.77	109.36	28.42	0.74	0.87	1.63
	K-means++	57.99	53.37	50.07	33.27	5.91	0.80	0.58	0.43	0.16	152.77	109.36	28.42	0.74	0.87	1.63
	K-means BISECT	53.92	53.37	49.63	27.57	5.91	0.17	0.59	0.43	0.16	135.35	109.36	25.68	0.70	0.87	1.74
	hierarchy	57.99	55.13	49.89	33.27	14.53	0.61	0.58	0.38	0.21	152.77	76.80	32.00	0.74	0.80	1.60
	Fuzzy C-means	58.79	54.51	49.89	34.29	7.52	0.54	0.58	0.42	0.15	151.94	108.00	25.77	0.75	0.88	1.72
	spectral	59.63	49.56	49.89	29.23	0.06	0.53	0.56	0.36	−0.01	150.64	58.40	1.18	0.77	1.01	7.72
explicit	DBSCAN (automated)	50.11	49.56	49.56	16.82	0.40	3.08	0.58	0.36	−0.24	42.20	39.50	1.26	1.17	1.44	2.09
	DBSCAN (manual)	49.74	49.56	50.33	16.55	0.65	1.18	0.62	0.37	−0.12	44.99	35.88	7.72	0.48	1.72	2.82
	HDBSCAN	49.69	50.84	50.18	14.62	2.92	2.30	0.59	0.26	−0.18	23.04	22.94	4.36	0.53	2.89	6.87
	mean shift	52.86	54.91	49.74	25.78	12.06	0.35	0.60	0.40	0.19	127.52	92.21	29.54	0.67	0.81	1.65
	OPTICS	71.96	49.67	49.52	34.06	0.60	12.19	0.18	0.34	0.04	33.86	33.75	0.55	1.42	1.55	1.11
	affinity	50.88	55.51	49.63	15.24	11.19	0.16	0.49	0.38	0.05	20.43	104.28	19.01	1.12	0.80	1.73
	BIRCH	49.98	49.98	49.98	0.90	0.90	0.90	−0.09	0.24	−0.09	17.98	51.25	17.98	2.03	1.29	2.03

https://doi.org/10.1021/acsomega.4c02502
ACS Omega 2024, 9, 21595−21611

http://pubs.acs.org/journal/acsodf

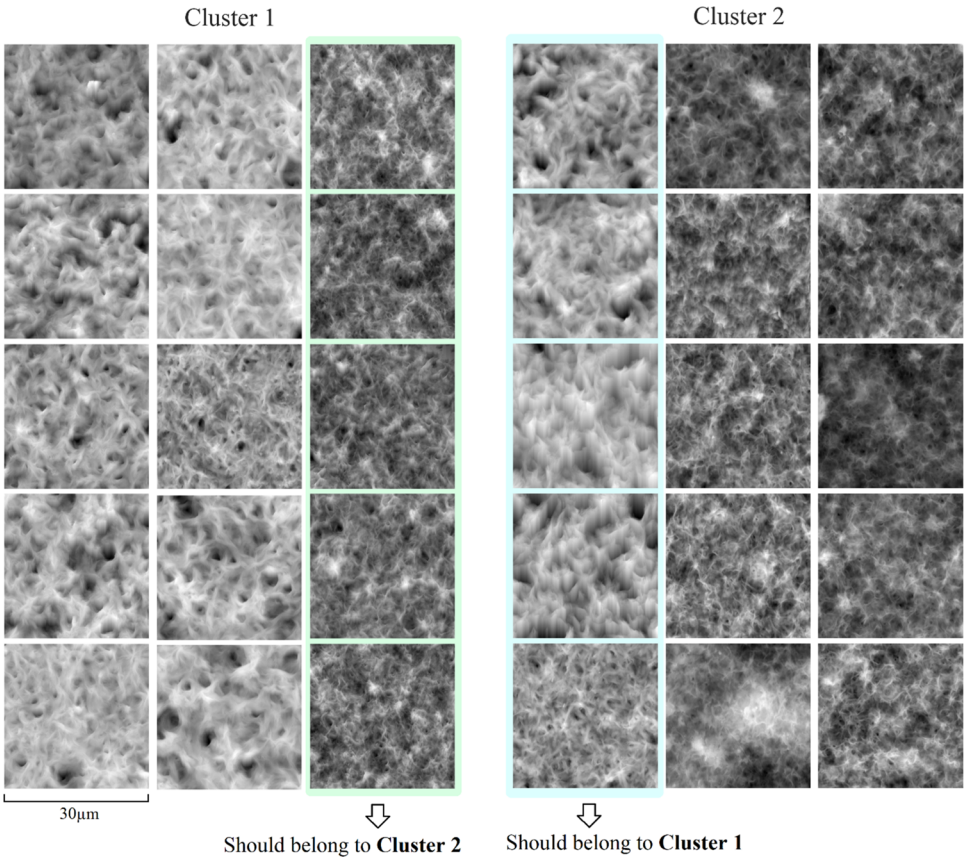


Figure 6. Results of clustering polyhydroxyalkanoate films by their type (P3HB and P3HB-*co*-3HV) when using data of a single scan size. Each image is 30 μm × 30 μm.

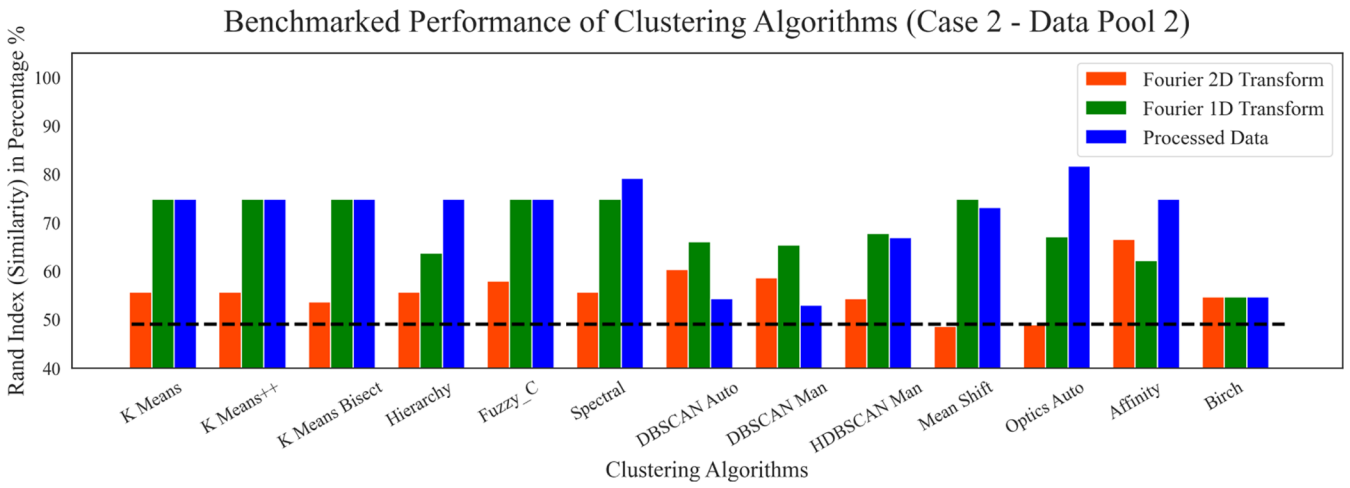


Figure 7. A graphical representation of prediction similarity (Rand Index) for a data pool consisting exclusively of 30 μm scan samples of P3HB and P3HB-*co*-3HV data when classified by the polymer type (case 2-B). The dotted line at 50% denotes the probability of a sample being assigned to a cluster by chance and the similarity index for such random clustering.

Optics algorithm using VD data, which classified P3HB-*co*-3HV samples into a single cluster but ignored the majority of P3HB as noise, thereby generating a high score.

Further, in this case, we did not observe differences between algorithms based on their type of tuning. This is largely attributed to demanding only two clusters as a result, which increases the chances of creating clusters by chance or noise over actual classification.

Case 3: Classification by Thickness (Threshold = 6). The expected results in this case are six clusters differentiated by their corresponding film thicknesses, as mentioned in Table S1. We created four data pools to maximize our insight into the classification of films based on their thicknesses attributed to varying the concentration of polymer powder during fabrication. Data pool A consisted of combined P3HB and P3HB-*co*-3HV of all scan sizes, pool B consisted of exclusively P3HB-*co*-3HV data of films 1 and 6, pool C consisted of P3HB data of scan size 30

https://doi.org/10.1021/acsomega.4c02502
ACS Omega 2024, 9, 21595−21611

μm from films 1, 3, and 6, last, pool D consisted of P3HB data of all sizes and film thicknesses.

On clustering and analyzing, there were instances when explicitly tuned algorithms such as Mean Shift, DBSCAN, and HDBSCAN did not converge to 6 clusters and produced 4−5 clusters as the resulting output. On observing M2 metrics, the overall mean variance between clusters was relatively high across all algorithms; however, the overlapping of elements between clusters was most observable in explicitly tuned algorithms. We compared M1 metrics to those of M2 and found that the accuracy of similarity (Rand Index) was unusually high compared to the remaining cases, and an example is shown in Table 4.

Despite the ground truth being vastly different from the prediction, the Rand Index score was high due to the permutation of pairs that are identical between the result and the ground truth, hence causing a disproportionately high score. To resolve this, we analyzed the M1 metrics via the Adjusted Rand Index that accounts for the possible permutations in the results, hereby showing prediction by chance within the score itself. The same was overlooked for prior cases (1 and 2) where we confirmed that the Adjusted Rand Index accepted the behavior indicated by M2 metrics.

In data pools A, B, and D, the Adjusted Rand Index ranged between negative values and 3, and this indicates a poor prediction of results when contrasted against the ground truth data.

Pool B showcases an example where the samples have been grouped according to their scan sizes of 5 and 10 μm in one cluster and 30 μm in another. Pool C consists of nine samples that were clustered similarly (i.e., based on their scan area over film thickness) shown in Figure 8. This was also visually observed in the prior data pools of A and D, where the samples were clustered according to their heights and surface features over thicknesses. The M2 metrics can be concluded as a positive outlook if deduced based on the classification done internally rather than by their thicknesses.

Case 4: Iterative Classification. In this case, the cluster threshold (the expected number of clusters) is set to reflect a sequence of specific features, such as the scan resolution, polymer type, or film thickness. Unlike previous cases where the clustering operation is performed on a pool of data only once, here upon completion of an iteration, the resulting elements are successively classified as per the following threshold creating subclusters. Based on the results from prior cases, we selected the iterative hierarchy as 3−2−6 related to the size of the scanned area and the type of polymer followed by the film thickness; Figure S3 provides an overview of the process. The data used in this case is a combined pool of both P3HB and P3HB-co-3HV of all scan sizes. In the first iteration, the clusters were differentiated from each other as per their scan area (5, 10, and 30 μm), the behavior and calculated metrics are as observed in Case 1 using data pool C. M2 metrics agreed with those of M1 in this iteration. Figure 9 shows the results of the hierarchy algorithm when clustering the data in the first iteration (by the size of the scanned area). The following iteration attempts to classify the elements of each subcluster as per their polymer type. Within all three clusters, we observed an imbalance where subcluster 1 had most elements in a single group and subcluster 2 had elements that did not fit within the latter, and this was observed in the Adjusted Rand Index score. The remaining elements in both subclusters were further classified into six groups attributed to the film thickness. On completing the

Table 4. Benchmarked Metrics of Case 3—Data Pool A Showcasing a High Rand Index Score Despite Having Incorrect Predictions versus the Ground Truth[a]

	metrics	Rand index (in %)			Adjusted Rand Index (in %)			Completeness (in %)			Silhouette score			Calinski−Harabasz index			Davies Bouldin index		
tuning	data transform	VD	1D	2D	VD	1D	2D	VD	1D	2D	VD	1D	2D	VD	1D	2D	VD	1D	2D
	algorithms																		
implicit	K-means	69.73	71.08	73.10	−0.01	−1.47	−0.19	11.07	6.78	9.48	0.14	0.41	−0.13	83.46	115.51	9.83	1.66	0.81	6.03
	K-means++	60.97	70.95	73.00	0.07	−1.52	0.16	12.47	6.72	9.12	0.29	0.42	−0.14	84.66	115.67	9.73	1.54	0.81	5.36
	K-means BISECT	62.60	70.86	73.70	0.19	−1.27	3.00	11.88	7.13	14.09	0.05	0.40	−0.15	77.16	110.09	7.94	1.34	0.82	4.97
	hierarchy	60.64	71.43	71.70	0.62	0.23	2.25	16.10	9.51	13.02	0.31	0.36	−0.21	83.26	101.03	9.69	1.51	0.82	4.04
	Fuzzy C-means	70.31	72.69	71.56	0.45	−0.57	1.06	11.14	7.82	11.17	0.15	0.38	−0.22	84.95	108.85	8.03	1.64	0.91	4.36
	spectral	73.59	72.22	72.73	2.78	−0.94	−1.18	10.71	6.58	5.23	0.01	0.38	−0.17	27.63	98.36	0.58	3.96	0.86	10.01
explicit	DBSCAN (automated)	24.87	49.65	21.17	0.17	−1.28	0.15	33.87	8.63	33.09	0.54	0.12	−0.40	15.12	21.39	0.44	0.42	0.59	1.47
	DBSCAN (manual)	21.06	67.95	42.00	0.11	−1.49	0.03	29.55	8.73	7.86	0.62	0.16	−0.21	44.99	28.41	1.68	0.48	2.34	9.21
	HDBSCAN	24.91	79.12	69.18	0.35	0.22	1.74	34.77	21.97	10.87	0.55	0.18	−0.16	42.82	15.43	8.98	1.16	2.02	5.22
	mean shift	31.94	70.93	53.46	0.16	−0.40	−1.07	9.79	8.77	11.54	0.60	0.37	−0.31	127.52	92.84	4.08	0.67	0.67	1.25
	OPTICS	26.96	63.24	17.33	−0.01	−0.54	0.11	16.23	11.03	34.96	−0.18	−0.07	0.04	20.74	11.77	0.55	1.03	1.69	1.11
	affinity	30.51	74.30	68.22	−0.10	−0.95	1.98	25.01	9.40	13.09	0.49	0.38	−0.08	20.43	104.28	16.89	1.12	0.80	5.70
	BIRCH	71.70	71.70	71.70	2.25	2.25	2.25	13.02	13.02	13.02	−0.21	0.01	−0.21	9.69	22.78	9.69	4.04	5.58	4.04

[a]The Adjusted Rand Index correlates to the performance of the M2 metrics with sparse overlapping of clusters.

https://doi.org/10.1021/acsomega.4c02502
ACS Omega 2024, 9, 21595−21611

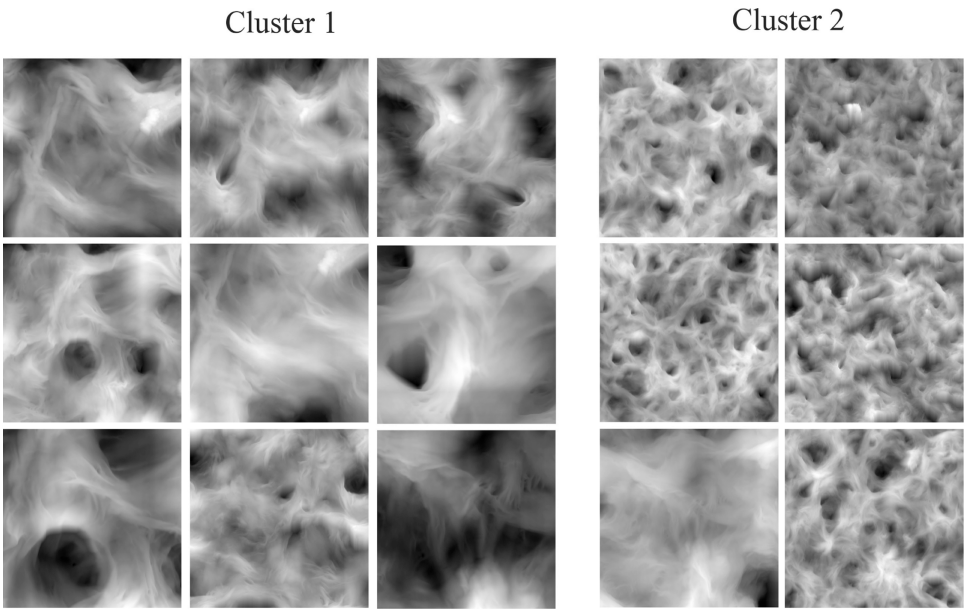


Figure 8. Visualized results of clustering a data pool consisting exclusively of P3HB-*co*-3HV films of thicknesses 1 and 6 of all sizes (5, 10, and 30 μm). The results show cluster 1 comprising scan data of sizes 5 and 10 μm, while cluster 2 comprises scan data of size 30 μm instead of being clustered by their film thicknesses.

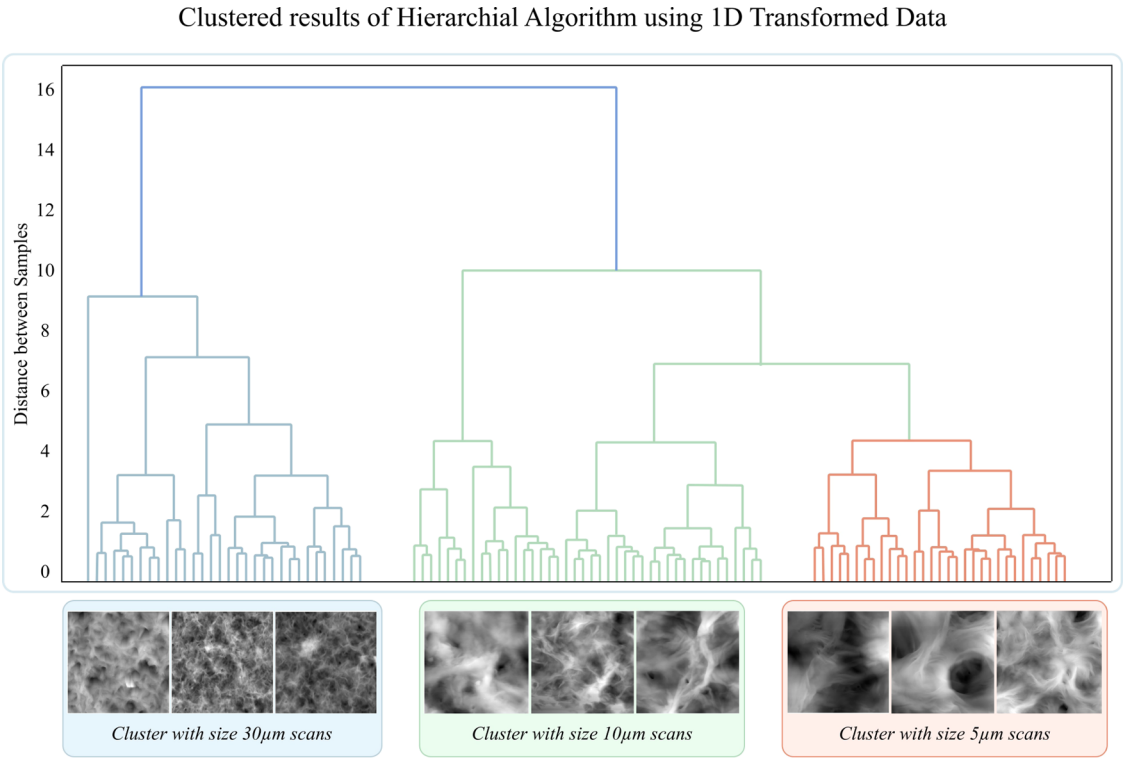


Figure 9. Graphical visualization of the distribution of elements when using 1D-FFT data via the Hierarchical clustering algorithm to classify elements based on their size of scanned area.

iteration, the M2 metrics indicated a high level of intercluster separation (variance between clusters) but the Adjusted Rand Index score was rather low. We investigated the results visually and observed that the elements were classified based on their surface resemblance, height differences, or underlying patterns over film thicknesses. An exceptional case was observed when trying to classify subcluster 2 of the 5 μm cluster. After the completion of the second iteration, the remaining number of elements was 5, whereas the designated threshold was 6; we reduced the threshold to match the thickness of the existing films (i.e., three clusters) to investigate the results and find the resulting elements to be grouped into individual clusters with high M2 metric scores indicating well-separated clusters. Figure S3 shows the results of the Fuzzy C-means algorithm when performing iterative clustering in the hierarchy of 3−2−6. Overall, by simulating Cases 1−4, we observed 1D-FFT

https://doi.org/10.1021/acsomega.4c02502
ACS Omega 2024, 9, 21595−21611

transform to have an improved performance compared to 2D-FFT and processed data (VD) across all algorithms collectively. The most accurate prediction was obtained when the elements were classified according to their scan area. Centroid-based algorithms had steady performance compared to density-based algorithms, which tended to classify a portion of elements as noise rather than into a cluster, thereby impacting the overall performance metrics, and this can be resolved by using adjusted metrics that consider the permutations of chances. Peculiar behavior was observed in the BIRCH algorithm, which had similar scores for all types of data transforms, and the DBSCAN automated algorithm had numerous instances where it did not converge to the required number of clusters, thereby producing low scores due to nonalignment of predicted and truth data. OPTICS had high scores in case 2 when using Processed data (VD), and this is mainly attributed to producing one well-defined cluster while labeling the rest as noise. In cases where self-evaluation metrics did not agree with the selected ground truth metrics (e.g., high Rand Index despite low silhouette score), we had to select adjusted metrics to analyze the results based on actual classification over clustering by chance. This approach was vital in drawing conclusions for case 3. We also observed that when the scores of the Rand Index were lower than the cutoff threshold by more than 5%, the clusters were overlapping.

Across all cases, we observed close accuracy variation between 1D-FFT and 2D-FFT relative to the algorithm being used. We found this to be attributed to the architecture of the algorithm used (i.e., centroid and density-based), while tuning of certain algorithms played a key role in shaping the outcome. Nevertheless, when attributed to the type of data used, 1D-FFT brings a layer of duality in the relative dimension (i.e., orientation by rows and columns), whereas data from 2D-FFT are resolved into a single component. For future full-scale unsupervised classification tasks, we believe the use of 2D-FFT can be a better choice over 1D-FFT as it overlooks the need for directional alignment at the cost of computational power and time. However, the algorithms used also play a key role in defining the self-evaluation criteria and variation of the accuracy. For example, when using 1D-FFT, using incorrect orientation (across the scan direction instead of along the scan direction) produces a significantly observable metric score (namely, a negative silhouette score and a low adjusted Rand index). In our cases of application, the overall difference between 1D-FFT vs 2D-FFT is marginal for most algorithms (not very significant + −5%), except in spectral. Spectral algorithm generates an affinity matrix across samples, with both orientations and height component being included, and the similarities across samples can coincide, making the affinity matrix saturated over being differentiable, therefore leading to a low score for 2D-FFT over 1D-FFT. We found 2D-FFT to be a more effective choice when using implicitly tuned algorithms and 1D-FFT marginally better than the latter for explicitly tuned algorithms except for DBSCAN and Mean shift algorithms, where the latter follows a top-down approach of splitting into samples. This way, 1D-FFT allows better differentiation in contrast to 2D-FFT that is based on combining either orientation. Mean shift follows the K-means approach of centroid-based clustering by assembling clusters around a heavy mean point; a similar relevance is observed as in 1D-FFT for DBSCAN.

DISCUSSION

From our benchmarking experiment, we concluded the performance of clustering algorithms, evaluation metrics, and the impact of using frequency domain data over applying clustering algorithms directly to the AFM data. In this section, we address topics that were encountered during our experiment and are to be considered in future works.

Frequency Domain of an Image. The Fourier theory states that a signal can be expressed as an infinite sum of sine waves. Discrete Fourier Transform involves a truncation of this infinite series. For images, the brightness level across the image can be considered as the signal. A single Fourier term encodes the spatial frequency, magnitude, and phase. Images obtained by the microscope are represented in the spatial domain, and the Fourier transformation converts the signal to the frequency domain.

An image normally consists of a 2D array of pixels. AFM data can be considered grayscale images with only one value per pixel or one channel. The image is defined by the intensity values at each spatial position. In the frequency domain, each image channel is represented in terms of sinusoidal waves: amplitude values that are stored in locations based not on *X*- and *Y*-spatial coordinates, but on *X*- and *Y*-frequencies. Since this is a digital representation, the values are discrete, the frequencies are multiples of the smallest or unit frequency, and the pixel coordinates represent the indices or integer multiples of this unit frequency.

Direction of Scan and 1D-FFT. When performing 1D-FFT, we extracted the scanned data in the orientations of the rows and columns. Within our experiment, we ultimately used the data extracted in the orientation of rows, which matches the direction of the actual AFM scan because the data extracted by columns produced results that were less accurate and, in some cases, produced vague results (i.e., the cluster elements seemed to be classified randomly). On investigating, we found that by matching the scan direction to the orientation of the transformed scan data, the results of clustering were positively impacted. Within 2D-FFT, the orientation is internally resolved when performing FFT, thereby interpreting the image as a 2D matrix over a line-by-line resolution as in 1D-FFT. However, this means that we treat the data as isotropic 2D images and ignore the scan direction, which may cause problems if some of the images in the data set have different orientations. Most AFM data, however, have the scan direction in the rows unless it is manually rotated after acquisition.

Performance of Cluster Architecture on High-Dimensional Data. Initially, it was assumed that centroid-based algorithms might perform poorly compared to density-based algorithms considering the multidimensional nature of the data, yet the performance was the opposite. Due to the fragile balancing of hyperparameters required for density-based algorithms, minute chances had huge effects on the resulting classification (e.g., considering all samples as noise or majority elements being grouped into a single cluster). We aim to extend and analyze the results of using multivariate data of small−medium dimensional nature on either group of algorithms in our future works.

Describing a Cluster without Descriptive Features. During our hybrid analysis of the results, we were able to observe many occurrences when certain AFM scans were classified into clusters incorrectly as per our truth data; however, when visualized, we found certain surface features that were common

https://doi.org/10.1021/acsomega.4c02502
ACS Omega 2024, 9, 21595−21611

within cluster elements (e.g., texture of certain samples, occurrence of physical features (for example crests or troughs)). From our simulated cases and investigation, we found the silhouette score and the Calinski−Harabasz index to have mutually agreeing scores and provide maximal insight into the structure of clusters, thereby aiding in differentiating variations within the data. In Case 1—Data Pool C and Case 2—Data Pool A, we observe distinct performance of the Silhouette score and the Calinski−Harabasz index, allowing differentiation between size (30 μm) scans against the rest and a P3HB against P3HB-*co*-3HV, respectively. This provides a crucial first-layer classification step that allows for the sectioning of the majority of elements from the rest and the performance of further operations as needed. This approach can be a viable choice when classifying huge polymer data sets that have a variety of polymer compounds with significant underlying patterns and correlations. We believe that there are underlying patterns that can be explored to reveal details that may have been overlooked during our current approach to classification. In our future work, we aim to incorporate descriptive and characteristic features during the clustering process, thereby boosting the overall accuracy of the results and the possibility to draw further conclusions.

CONCLUSIONS

The benchmarked performance of the clustering algorithms provides insight into the use of high-dimensional data for clustering and showcases the impact of using various data transformations to classify surface scans of polyhydroxyalkanoate films. The accuracy of classification is measured using ground truth and self-evaluation metrics. Our results indicate that when using a vectorized approach to classify scan data, the best-performing results were produced by applying the 1D Fourier transform on the data, which produced an overall accuracy of 75.29% across all algorithms and data pools when classifying by the size of the scanned area. The K-means, K-means++, Hierarchy, and Fuzzy C-means algorithms had similar performance, with instances of K-means bisect having slightly lower accuracy compared to the rest. However, explicitly tuned algorithms were prone to designating elements as noise over clusters, and this was attributed to sensitive tuning at high dimensionality. The DBSCAN automated algorithm had the lowest score due to nonconvergence similarly observed in HDBSCAN.

Our hybrid approach can be validated when investigating the results of clustering for benchmarking; we were able to relate self-evaluation metrics with the ground truth metrics and have highlighted the usage of permutation-adjusted metrics in certain cases. We highlight the relevance and effectiveness of using 1D and 2D Fourier transforms relative to the architectural type of algorithms, with 2D Fourier transformed data being more suitable for centroid-based algorithms and 1D Fourier transform being an effective choice for explicitly tuned algorithms. The results of this study and the developed toolkit are publicly available on GitHub, and the annotated data set used for benchmarking is available on Zenodo database. In future works, we aim to extend our toolkit and improve our existing case of clustering AFM polymer scans by including in-depth characteristic information that provides better individuality to the data (e.g., Roughness, Young's modulus, etc.) and therefore improve the overall results of classification and investigation.

ASSOCIATED CONTENT

Data Availability Statement

The data set of P3HB and P3HB-CO-3HV scanned via the AFM is available for open access on Zenodo database (10.5281/zenodo.10621270). The code used for this experiment for simulations and the toolkit are available at https://github.com/ITMO-MMRM-lab/MicroClust/releases/tag/V1.

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acsomega.4c02502.

Film thicknesses of P3HB and P3HB-CO-P3HV polymers measured using a digital micrometer after synthesis; benchmarked performance metrics of clustering algorithms when classifying a data pool consisting of exclusively P3HB data by their scanned size; graphical representation of prediction similarity (Rand index) for data pool B (above) consisting exclusively of P3HB-CO-P3HV data and data pool C (below) consisting of both P3HB and P3HB-CO-P3HV data combined; and visualized representation of the iterative clustering approach followed in Case 4 simulations to classify the scanned data as per iterative hierarchy (PDF)

AUTHOR INFORMATION

Corresponding Authors

Ashish T. S. Ireddy − *Infochemistry Scientific Centre, ITMO University, 191002 St. Petersburg, Russia*; Email: ireddy@itmo.ru

Michael Nosonovsky − *Infochemistry Scientific Centre, ITMO University, 191002 St. Petersburg, Russia; University of Wisconsin-Milwaukee, Milwaukee, Wisconsin 53217, United States*; orcid.org/0000-0003-0980-3670; Email: nosonovs@uwm.edu

Pavel S. Zun − *Infochemistry Scientific Centre, ITMO University, 191002 St. Petersburg, Russia*; Email: pavel.zun@itmo.ru

Authors

Fares D. E. Ghorabe − *Infochemistry Scientific Centre, ITMO University, 191002 St. Petersburg, Russia*

Ekaterina I. Shishatskaya − *Infochemistry Scientific Centre, ITMO University, 191002 St. Petersburg, Russia*

Galina A. Ryltseva − *Siberian Federal University, 660041 Krasnoyarsk, Russia*

Alexey E. Dudaev − *Siberian Federal University, 660041 Krasnoyarsk, Russia*

Dmitry A. Kozodaev − *NT-MDT BV, 7335 Apeldoorn, The Netherlands*

Ekaterina V. Skorb − *Infochemistry Scientific Centre, ITMO University, 191002 St. Petersburg, Russia*; orcid.org/0000-0003-0888-1693

Complete contact information is available at: https://pubs.acs.org/10.1021/acsomega.4c02502

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

The authors acknowledge the Ministry of Science and Higher Education (Project FSER-2024-0003). Priority 2030 Program is acknowledged for Infrastructural Support.

■ ADDITIONAL NOTES

[1]https://github.com/ITMO-MMRM-lab/MicroClust/releases/tag/V1
[2]https://github.com/ITMO-MMRM-lab/MicroClust/releases/tag/V1

■ REFERENCES

(1) Young, R. J.; Lovell, P. A. *Introduction to Polymers*; CRC Press, 2011.
(2) Biron, M. *Thermoplastics and Thermoplastic Composites*; William Andrew, 2018.
(3) Chanprateep, S. Current trends in biodegradable polyhydroxyalkanoates. *J. Biosci. Bioeng.* **2010**, *110* (6), 621−632.
(4) Chen, G.-Q.; Jiang, X.-R.; Guo, Y. Synthetic biology of microbes synthesizing polyhydroxyalkanoates (PHA). *Synth. Syst. Biotechnol.* **2016**, *1* (4), 236−242.
(5) Koller, M.; Mukherjee, A. "Polyhydroxyalkanoates−linking properties, applications, and end-of-life options. *Chem. Biochem. Eng. Q.* **2020**, *34* (3), 115−129, DOI: 10.15255/CABEQ.2020.1819.
(6) Gigante, V.; Cinelli, P.; Seggiani, M.et al.Processing and Thermomechanical Properties of PHA. In *Handbook of Polyhydroxyalkanoates*; CRC Press, 2020; pp 91−118.
(7) Shumilova, A. A.; Myltygashev, M. P.; Kirichenko, A. K.; et al. Porous 3D implants of degradable poly-3-hydroxybutyrate used to enhance regeneration of rat cranial defect. *J. Biomed. Mater. Res., Part A* **2017**, *105* (2), 566−577.
(8) Lizarraga-Valderrama, L. R.; Nigmatullin, R.; Taylor, C.; et al. Nerve tissue engineering using blends of poly (3-hydroxyalkanoates) for peripheral nerve regeneration. *Eng. Life Sci.* **2015**, *15* (6), 612−621, DOI: 10.1002/elsc.201400151.
(9) Williams, S. F.; Martin, D. P. Applications of polyhydroxyalkanoates (PHA) in medicine and pharmacy. In *Biopolymers*; Wiley, 2005.
(10) Rentsch, C.; Rentsch, B.; Breier, A.; et al. Evaluation of the osteogenic potential and vascularization of 3D poly (3) hydroxybutyrate scaffolds subcutaneously implanted in nude rats. *J. Biomed. Mater. Res., Part A* **2010**, *92A* (1), 185−195, DOI: 10.1002/jbm.a.32314.
(11) Nair, L. S.; Laurencin, C. T. Biodegradable polymers as biomaterials. *Prog. Polym. Sci.* **2007**, *32* (8−9), 762−798.
(12) Shishatskaya, E. I.; Volova, T. G.; Puzyr, A. P.; et al. Tissue response to the implantation of biodegradable polyhydroxyalkanoate sutures. *J. Mater. Sci.: Mater. Med.* **2004**, *15*, 719−728.
(13) Sun, J.; Wu, J.; Li, H.; et al. Macroporous poly (3-hydroxybutyrate-co-3-hydroxyvalerate) matrices for cartilage tissue engineering. *Eur. Polym. J.* **2005**, *41* (10), 2443−2449.
(14) Sanders, W. C. *Atomic Force Microscopy: Fundamental Concepts and Laboratory Investigations*; CRC Press, 2019.
(15) Hennig, C.; Meila, M.; Murtagh, F.; Rocci, R. *Handbook of Cluster Analysis*; CRC press, 2015.
(16) Fotiadis, D.; Scheuring, S.; Müller, S. A.; Engel, A.; Müller, D. J. Imaging and manipulation of biological structures with the AFM. *Micron* **2002**, *33*, 385−397.
(17) Nguyen-Tri, P.; Ghassemi, P.; Carriere, P.; Nanda, S.; Assadi, A. A.; Nguyen, D. D. Recent applications of advanced atomic force microscopy in polymer science: A review. *Polymers* **2020**, *12*, No. 1142, DOI: 10.3390/polym12051142.
(18) Zhukov, M.; Hasan, M. S.; Nesterov, P.; Sabbouh, M.; Burdulenko, O.; Skorb, E. V.; Nosonovsky, M. Topological data analysis of nanoscale roughness in brass samples. *ACS Appl. Mater. Interfaces* **2022**, *14*, 2351−2359.
(19) Aglikov, A. S.; Aliev, T. A.; Zhukov, M. V.; Nikitina, A. A.; Smirnov, E.; Kozodaev, D. A.; Nosonovsky, M. I.; Skorb, E. V. Topological Data Analysis of Nanoscale Roughness of Layer-by-Layer Polyelectrolyte Samples Using Machine Learning. *ACS Appl. Electron. Mater.* **2023**, *5*, 6955−6963.
(20) Xing, F.; Xie, Y.; Su, H.; Liu, F.; Yang, L. Deep Learning in Microscopy Image Analysis: A Survey. *IEEE Trans. Neural Networks Learn. Syst.* **2018**, *29*, 4550−4568.
(21) Bolshakova, A. V.; Kiselyova, O. I.; Yaminsky, I. V. Microbial surfaces investigated using atomic force microscopy. *Biotechnol. Prog.* **2004**, *20*, 1615−1622.
(22) Alldritt, B.; Hapala, P.; Oinonen, N.; Urtev, F.; Krejci, O.; Canova, F. F.; Kannala, J.; Schulz, F.; Liljeroth, P.; Foster, A. S. Automated structure discovery in atomic force microscopy. *Sci. Adv.* **2020**, *6*, No. eaay6913, DOI: 10.1126/sciadv.aay6913.
(23) Zhukov, M. V.; Aglikov, A. S.; Sabboukh, M.; Kozodaev, D. A.; Aliev, T. A.; Ulasevich, S. A.; Nosonovsky, M.; Skorb, E. V. AFM-Topological Data Analysis of Brass after Ultrasonic Surface Modification. *ACS Appl. Eng. Mater.* **2023**, *1*, 2084−2091.
(24) Alsteens, D.; Muller, D. J.; Dufrene, Y. F. Multiparametric atomic force microscopy imaging of biomolecular and cellular systems. *Acc. Chem. Res.* **2017**, *50*, 924−931.
(25) Fang, S. J.; Haplepete, S.; Chen, W.; Helms, C.; Edwards, H. Analyzing atomic force microscopy images using spectral methods. *J. Appl. Phys.* **1997**, *82*, 5891−5898.
(26) Mincheva, R.; Raquez, J.-M.The Surface of Polymers. In *Surface Modification of Poly- mers: Methods and Applications*; Wiley, 2019.
(27) Sadullah, M. S.; Xu, Y.; Arunachalam, S.; et al. Predicting droplet detachment force: Young-Dupré Model Fails, Young-Laplace Model Prevails. *Commun. Phys.* **2024**, 7 (1), No. 89, DOI: 10.1038/s42005-024-01582-0.
(28) Tadmor, R.; Das, R.; Gulec, S.; et al. Solid−liquid work of adhesion. *Langmuir* **2017**, *33* (15), 3594−3600.
(29) Tadmor, R. Open problems in wetting phenomena: pinning retention forces. *Langmuir* **2021**, *37* (21), 6357−6372.
(30) Volova, T. G.; Shishatskaya, E. I. Bacteria Strain Cupriavidus Eutrophus VKPM B-10646 - Producer of Polyhydroxyalka-Noates and Method of their Production. RF Patent RF2,439,143, 2012.
(31) Volova, T.; Kiselev, E.; Nemtsev, I.; et al. Properties of degradable polyhydroxyalkanoates with different monomer compositions. *Int. J. Biol. Macromol.* **2021**, *182*, 98−114.
(32) Volova, T. G.; Kiselev, E. G.; Shishatskaya, E. I.; et al. Cell growth and accumulation of polyhydroxyalkanoates from CO2 and H2 of a hydrogen-oxidizing bacterium, Cupriavidus eutrophus B-10646. *Bioresour. Technol.* **2013**, *146*, 215−222.
(33) Sharma, V.; Sehgal, R.; Gupta, R. Polyhydroxyalkanoate (PHA): properties and modifications. *Polymer* **2021**, *212*, No. 123161.
(34) Schlegel, H.; Kaltwasser, H.; Gottschalk, G. A submersion method for culture of hydrogen-oxidizing bacteria: growth physiological studies. *Arch. Mikrobiol.* **1961**, *38*, 209−222.
(35) Nečas, D.; Klapetek, P. Gwyddion: an open-source software for SPM data analysis. *Open Phys.* **2012**, *10*, 181−188.
(36) Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V. others Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825−2830.
(37) Xu, R.; Wunsch, D. Survey of clustering algorithms. *IEEE Trans. Neural Networks* **2005**, *16*, 645−678.
(38) Omran, M.; Engelbrecht, A.; Salman, A. An overview of clustering methods. *Intell. Data Anal.* **2007**, *11*, 583−605, DOI: 10.3233/IDA-2007-11602.
(39) Hamerly, G.; Elkan, C. In *Alternatives to the k-Means Algorithm that Find Better Clusterings*, Proceedings of the Eleventh International Conference on Information and Knowledge Management; ACM Digital Library, 2002; pp 600−607.
(40) Arthur, D.; Vassilvitskii, S. In *K-means++ the Advantages of Careful Seeding*, Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms; ACM Digital Library, 2007; pp 1027−1035.
(41) Steinbach, M.; Karypis, G.; Kumar, V. A Comparison of Document Clustering Techniques 2000 https://conservancy.umn.edu/bitstream/handle/11299/215421/00-034.pdf.
(42) Dias, M. L. D. fuzzy-c-means: An implementation of Fuzzy C-means clustering algorithm, 2019. https://git.io/fuzzy-c-means.
(43) Meila, M.Spectral Clustering: A Tutorial for the 2010's. In *Handbook of Cluster Analysis*; CRC Press, 2016.

https://doi.org/10.1021/acsomega.4c02502
ACS Omega 2024, 9, 21595−21611

(44) Ester, M.; Kriegel, H.-P.; Sander, J.; Xu, X.et al.In *A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise*; kdd., 1996; pp 226−231.
(45) Rahmah, N.; Sitanggang, I. S. Determination of optimal epsilon (eps) value on DBscan algorithm to clustering data on peatland hotspots in Sumatra. *IOP Conf. Ser.: Earth Environ. Sci.* **2016**, *31*, No. 012012, DOI: 10.1088/1755-1315/31/1/012012.
(46) Campello, R. J. G. B.; Moulavi, D.; Sander, J. In *Density-Based Clustering Based on Hierarchical Density Estimates*, Pacific-Asia Conference on Knowledge Discovery and Data Mining; Springer, 2013; pp 160−172.
(47) Comaniciu, D.; Meer, P. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 603−619, DOI: 10.1109/34.1000236.
(48) Ankerst, M.; Breunig, M. M.; Kriegel, H.-P.; Sander, J. OPTICS: Ordering points to identify the clustering structure. *ACM SIGMOD Rec.* **1999**, *28*, 49−60.
(49) Frey, B. J.; Dueck, D. Clustering by passing messages between data points. *Science* **2007**, *315*, 972−976.
(50) Zhang, T.; Ramakrishnan, R.; Livny, M. BIRCH: an efficient data clustering method for very large databases. *ACM SIGMOD Rec.* **1996**, *25*, 103−114.
(51) Rokach, L.; Maimon, O.Clustering methods. In *Data Mining and Knowledge Discovery Handbook*; Springer, 2005; pp 321−352.
(52) Romano, S.; Vinh, N. X.; Bailey, J.; Verspoor, K. Adjusting for chance clustering comparison measures. *J. Mach. Learn. Rese.* **2016**, *17*, 4635−4666.
(53) Hubert, L.; Arabie, P. Comparing partitions. *J. Classif.* **1985**, *2*, 193−218.
(54) Rosenberg, A.; Hirschberg, J. In *V-measure: A Conditional Entropy-Based External Cluster Evaluation Measure*, Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL); Association for Computational Linguistics, 2007; pp 410−420.

https://doi.org/10.1021/acsomega.4c02502
ACS Omega 2024, 9, 21595−21611