Developing the Computational Image Processing Method for Quantitative Analysis of Nanopore Structure Obtained from HS-AFM (AFMnanoQ)

Thuyen Vinh Tran Nguyen[1,2], Ngoc Quoc Ly[1,2], Ngan Thi Phuong Le[1,2], Hoang Duc Nguyen[1,2], Kien Xuan Ngo[3]

[1] University of Science, VNUHCM, Ho Chi Minh, Vietnam
[2] Viet Nam National University Ho Chi Minh City, Ho Chi Minh City, Vietnam
[3] WPI-Nano Life Science Institute, Kanazawa University, Japan
vinhthuyen7494@gmail.com, lqngoc@fit.hcmus.edu.vn, ngoxuankien@staff.kanazawa-u.ac.jp

Abstract. High-Speed Atomic Force Microscopy (HS-AFM) enables nanoscale imaging of biological structures with exceptional spatial (1 nm in the x-y plane; ~0.1 nm in the z-direction) and temporal resolution (~20 ms per frame). HS-AFM encodes three-dimensional (3D) information within a two-dimensional (2D) format, where the lateral dimensions (x, y) of structures correspond to spatial positioning in the image, while height (z) information is embedded in pixel intensity. This unique data structure presents significant challenges in segmentation and morphological analysis, requiring specialized computational approaches. To overcome these limitations, we develop "AFMnanoQ", a feature-driven computational framework for segmentation and morphological measurement of HS-AFM data. Our method operates independently of labeled training data, making it robust to data scarcity while simultaneously serving as a powerful tool for generating high-quality labeled datasets for future deep-learning applications. We validate AFMnanoQ using both synthetic and experimental AFM/HS-AFM datasets, including semi-automatic analyses of the conformations and dynamics of alpha-hemolysin (αHL), a β-barrel pore-forming toxin (PFT) secreted by *Staphylococcus aureus*. Our method achieves competitive performance with deep-learning models while maintaining superior adaptability across diverse HS-AFM datasets. As the future perspective, we plan to further develop or integrate it with deep-learning models to enhance segmentation performance and to reconstruct 3D structures from experimental AFM images. This will leverage conformational libraries generated in this study, enabling cross-validation between the two methods and ultimately bridging the gap between feature-driven and data-driven approaches in AFM image analysis.

Keywords: AFMnanoQ, AFM, HS-AFM, Segmentation, Morphological Parameters, Geometric Measurement, Computer Vision, Alpha-hemolysin (αHL), Pore-Forming Toxin (PFT), Nanopore, *Staphylococcus aureus*.

1 Introduction

High-Speed Atomic Force Microscopy (HS-AFM)[1] offers nanometer resolution and real-time imaging, making it ideal for monitoring alpha-hemolysin (αHL) assembly. However, HS-AFM images exhibit significant noise and heterogeneity across different experimental conditions, posing challenges for automated segmentation and quantitative analysis.

It is apparent that inherent upper limits exist in the instrumental advancements of HS-AFM[1]. In general, AFM data provide 2D surface information with a spatial resolution of approximately 1 nm in the x-y plane and height information with a resolution of around 0.1 nm in the z-direction and temporal resolution of milliseconds to minutes. To mitigate these limitations in spatial and temporal resolutions of experimental AFM data, various theoretical groups have made noteworthy contributions. Their objective is to enhance the resolutions of HS-AFM, facilitating a more comprehensive study of protein dynamics and atomic structures. Despite the persistent limitations of rigid fitting in localization AFM[2], NanoLocz[3], AMFmize[4], and simulation AFM[5][6], these approaches have paved the new ways to transform 2D surface experimental AFM images into 3D atomic conformations.

Simulation AFM models predominantly leverage existing protein databank (PDB)[1], transforming them into simulated AFM image structures (i.e., pseudo-AFM). However, validation of these simulated AFM models primarily hinges on the pixel maxima principle. The pixel resolution in these models predominantly relies on the tip apex radius, tip architecture (cone angle), and scanning step. Localization AFM attempts to enhance both the spatial and temporal resolutions of HS-AFM through the utilization of single-line scanning (x-scanning) over time[2][3]. However, these approaches only provide local pixel maxima derived from low-resolution 2D surface topography, without the incorporation of force-fields for accurately validating 3D fitted molecular structures. Force fields are crucial for accurately validating 3D fitted molecular structures, as they account for intermolecular interactions and energy minimization[7][8]. Without incorporating force fields, approaches relying solely on local pixel maxima from low-resolution 2D surface data may lead to less reliable 3D structural interpretations.

To overcome the limitation in spatial resolution of 2D AFM images, AFMfit, which utilizes an iterative fitting process that progressively modifies an input 3D atomic model (e.g., PDB) to match the 2D molecular surface images captured from multiple experimental AFM observations. AFMfit simultaneously examines the rigid and flexible degrees of freedom by employing the fast Fourier transform and the nonlinear rigid block normal-mode analysis (NOLB-NMA) method, respectively[9],[10]. While other flexible fitting approaches for AFM images have been proposed[11], AFMfit stands out for incorporating thorough validation, which is often lacking in other methods. Advancing in this direction, a recent study successfully integrated localization AFM (LAFM) with MD simulations to produce large-scale 3D conformations and density maps from 2D AFM data[12].

The introduction of alphafold 2 and 3 have revolutionized the protein structure prediction at the level of single atomic conformation and now has extended its capabilities

[1] RCSB PDB (https://www.rcsb.org/), PDBJ (https://pdbj.org/)

to predict the complex structures involving proteins, nucleic acids, small molecules, ions, and modified residues[13][14][15][16]. While deep learning has demonstrated as the state-of-the-art performance in image segmentation, it relies heavily on large, well-annotated datasets and requires retraining to adapt to different imaging conditions.

To overcome some of those limitations, we propose a feature-driven computational framework for segmentation and morphological measurement of HS-AFM data for analyzing αHL structures[17][18][19][20]. Our approach operates independently of labeled training data, ensuring adaptability across diverse HS-AFM datasets while also generating high-quality annotations for future deep-learning applications.

The framework comprises three main stages. The first stage addresses noise removal for the two main noise types in HS-AFM topographic images: (i) point and stripe noise and (ii) distortions[21]. We use Non-local Mean[22] and DeStripe[23] filters to mitigate these issues. Our denoising is flexibly calibrated to eliminate unwanted noise while avoiding over-filtering that could introduce artifacts or remove critical structural details. The second stage focuses on kernel detection, recognizing that the "kernel" may vary depending on the developmental stage of αHL structures, we introduce a three-step detection protocol. First, Blob Detection via the Hessian matrix identifies potential blob-like regions that may represent "kernel". Second, Blob Verification, we leverage the "mushroom-like" shape of αHL and the orthogonal, top-down imaging capture configuration of HS-AFM (which encodes height in color channels) to perform multi-level set clustering based on height. For each blob detected earlier, we check if its center is consistently encircled by near-circular or arc-shaped boundaries across different height levels. Blobs that meet this criterion are confirmed as valid kernels. Third, Curvature Boundary Extraction, we refine the segmentation by delineating precise boundaries at the pixel level through curvature-based methods. In the final stage, we extract and quantify key morphological parameters including perimeter, area, volume, and height using derivative-based calculations and integration, providing a comprehensive quantitative description of αHL's conformations.

2 Related Work

In this section, we review existing approaches for denoising and segmentation of AFM images, both traditional computational techniques and deep learning-based methods to highlight their respective strengths and limitations.

2.1 Denoising

AFM relies on cantilever interacting with the sample surface to capture images. This imaging modality is inherently prone to two primary types of noise: point and stripe noise and distortions[21]. Point noise appears as isolated pixels or "grains," while stripe noise manifests as linear artifacts across the image. In contrast, the distortions are scanning-related and generally appear as scan lines. To address these challenges, extensive research has been dedicated to denoising techniques, both *traditional computational methods* and *deep learning–based* approaches.

Traditional computational denoising methods offer a variety of approaches for removing noise while preserving structural details. *Morphology-based*[24] techniques (e.g., erosion, dilation, opening, and closing) remove isolated noise pixels yet may sometimes erode fine details. The *Median filter*[24] effectively reduces impulse noise

by replacing each pixel with the median of its neighborhood, maintaining edge sharpness. *Non-Local Means (NLM)*[22] filter averages similar pixels across the image—even if they are distant—to lower noise while retaining critical features. Although *Gaussian filters*[24] are computationally efficient, they tend to blur important edges and details. *DeStripe*[23], design specifically for stripe noise, filters the image in the frequency domain to subtract low-frequency stripe patterns with minimal impact on overall content. *Wavelet-based denoising*[25] decomposes the image into frequency components and uses soft thresholding to eliminate noise while preserving sharp transitions. Lastly, *Low-Rank Recovery*[26] models the image as a sum of a low-rank (clean) component and a sparse (noise) component, effectively isolating structured noise such as stripes, though it may be computationally intensive.

Table 1. Evaluation table of *traditional computational methods*.

Method	Effective denoising target	Data preservation
Morphology-Based Methods	Point	No (details maybe loss)
Median Filter	Point	Yes
Non-Local Means (NLM) Filter	Point	Yes
Gaussian Filter	Point	No (edges maybe loss)
DeStripe	Stripe	Yes
Wavelet-Based Denoising	Stripe	Yes (high cost, manual threshold)
Low-Rank Recovery	Stripe	Yes (high cost)

Deep learning–based methods have also been applied to AFM denoising. *MPRNet*[27] employs multi-stage architecture with residual learning and multi-scale fusion to remove complex noise while preserving fine details. *HINet*[27] leverages hierarchical multi-scale feature extraction for robust denoising. *Uformer*[27] uses a U-shaped transformer with self-attention to capture long-range dependencies and address both point and stripe noise. *Restormer*[27], optimized with multi-head self-attention for reduced complexity, excels at eliminating noise while maintaining structural integrity. According to [27], *Restormer* is the most effective, while *HINet* is the most efficient for denoising AFM images.

Traditional methods are computationally efficient, interpretable, and require no labeled training data, which enables real-time processing and explicit control over noise reduction and detail preservation. However, these methods can sometimes over-smooth AFM images, risking the loss of critical nanoscale features in the presence of complex noise patterns. In contrast, deep-learning approaches can learn complex, non-linear mappings to effectively remove both point and stripe noise, yet they demand large, high-quality training datasets and substantial computational resources. Within the scope of this study, the focus is on exploring the potential of traditional methods. In future work, deep-learning techniques will be experimentally evaluated and integrated with the traditional approaches presented here to further enhance image processing performance.

2.2 Segmentation

The goal of segmentation in AFM images is to accurately delineate objects of interest. This study examines two categories of segmentation methods: traditional computational methods and deep learning–based approaches.

Traditional segmentation techniques rely on explicit mathematical operations and heuristics.

- *Edge-Based Methods[24]*: These techniques detect intensity gradients using operators like Sobel or Canny to identify object boundaries. While computationally simple and interpretable, edge-based methods are highly sensitive to the noise and low contrast typical of AFM images, which can cause fragmented or incomplete boundaries.
- *Watershed[24]*: By interpreting the image as a topographic surface, the watershed algorithm "floods" regions from local minimum, with the meeting points of the floodwaters defining object boundaries. Although effective in delineating contiguous regions, the method is prone to over-segmentation in noisy images. Marker-controlled approaches can help mitigate this issue but require careful parameter selection.
- *Blob Detection[28]*: Techniques based on the Hessian matrix or Laplacian of Gaussian (LoG) detect regions (or "blobs") that correspond to the central parts of objects. However, variability in object shape and intensity levels necessitates robust configuration setting to ensure accurate segmentation.

Deep-learning models learn hierarchical features from data, often yielding state-of-the-art segmentation performance when sufficient annotated samples are available.

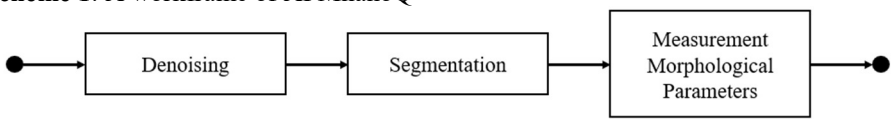
- *SegNet[29]*: uses encoder-decoder architecture to produce pixel-wise segmentation maps. It captures spatial hierarchies through convolutional layers, allowing for robust feature extraction even in complex and noisy environments. However, its effectiveness depends on the availability of large and annotated datasets.
- *Unet[30]*: has architecture, featuring skip connections, effectively merges low-level spatial details with high-level semantic information. This enables precise boundary delineation, making it particularly suitable for AFM images where fine structural details are critical. Nonetheless, like SegNet, its performance is heavily relied on ample training data.
- *YOLOv8[31]*: Though primarily designed for real-time object detection, YOLOv8 has been adapted for segmentation tasks, offering rapid instance-level segmentation. Its efficiency and ability to handle high-resolution images make it a compelling option for AFM data. However, its reliance on extensive labeled data can pose challenges when interpretability and precise nanoscale measurements are essential.

In comparison, traditional segmentation methods are efficient, transparent, and offer direct control over parameters, making them suitable for AFM image processing though they often need extensive tuning to adapt to specific needs. In contrast, deep-learning approaches achieve superior segmentation through learned representations but require large datasets, significant computational resources, and can be opaque in their decision-making. Balancing these trade-offs is essential for effective segmentation in AFM images. Our framework leverages these insights to develop a method tailored to our specific problem.

3 Method

Our computational framework (AFMnanoQ) addresses three core tasks: denoising, segmentation, and measurement of morphological parameters (**Scheme 1**). To validate the methodology, AFMnanoQ analyzed the conformational dynamics of the αHL toxin, which forms heptameric pores in a lipid membrane containing cholesterol, using HS-AFM imaging.

Scheme 1: A workframe of AFMnanoQ



Dataset

The dataset utilized in this study is currently unpublished[32]. The αHL[17][18][19] is a key virulence factor secreted by *Staphylococcus aureus* that disrupts host cell membranes by forming transmembrane pores. Despite extensive research, no targeted therapy exists, highlighting the need for improved diagnostics, prevention, and vaccine strategies[20]. Understanding the oligomerization process of αHL from individual monomers to fully assembled pores is therefore essential for vaccination and therapeutic targets.

Specifically, the dataset (**Fig. 1**) is organized into three distinct categories:

(i) **Intermediate Structures**: This includes monomers, dimers, trimers, tetramers, pentamers, or hexamers formation during the oligomerization process (**Fig. 1A**).

(ii) **Pre-Pore Structures**: Characterized by mushroom-like structures of heptamers formed on membrane surface with a height of approximately 10 nm (**Fig. 1B, D**).

(iii) **Pore Structures**: Similar to pre-pore structures, but with the transmembrane domain inserted into the lipid membrane, causing the height to decrease from 10 nm to around 6 nm in AFM images (**Fig. 1C**).

All HS-AFM image series in the dataset are stored in ASD format and are analyzed by the UMEX Viewer software, developed by Kenichi Umeda (Kanazawa University).

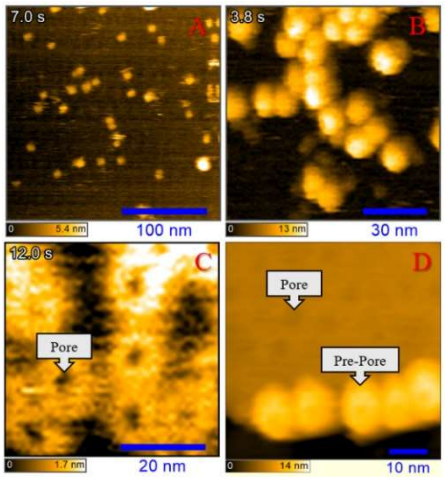


Fig. 1. Representative HS-AFM images from dataset illustrating developmental stages of αHL. (**A**) ***Intermediate Structures***; (**B**) ***Pre-Pore Structures***; (**C**) ***Pore Structures***; (**D**) ***Pre-Pore and Pore Structures.*** The elapsed time, lateral (x-y) and height (z) color-scale bars are shown. Top-left displays the elapsed time (in seconds) from the start of the imaging sequence, enabling temporal correlation of dynamic events. Bottom-left shows the vertical (height) color-scale bar, which converts the color-coded height data into nanometers for precise vertical measurements. Bottom-right presents the lateral (length and width) scale bar, translating pixel dimensions into nanometers to accurately quantify lateral dimensions.

To standardize the processing workflow, HS-AFM images are exported from ASD to PNG format with embedded metadata, ensuring accurate real-world measurements during analysis.

Problem Definition

The primary challenge lies in accurately segmenting the objects of interest, which may be individual monomers, the mushroom-shaped[18][19] regions corresponding to pre-pore structures, or fully formed pores from the HS-AFM images and subsequently extracting reliable morphological parameters of the objects in AFM images.

For each dataset corresponding to different developmental phases of αHL, the objectives are as follows:

- *Intermediate Structures*: segment individual monomers and clusters to compute morphological features such as perimeter, area, and height. These metrics are then used to estimate oligomer volumes during the oligomerization process.
- *Pre-Pore Structures*: focus on segmenting the characteristic mushroom-shaped region of αHL. The height information of these structures is computed to evaluate the progression of membrane penetration.
- *Pore Structures*: isolate the pore regions after membrane penetration and calculate key morphological parameters including perimeter, area, volume, and pore depth to elucidate the structural transition from pre-pore to pore states.

Mathematically, let us denote the HS-AFM image sequence as follows:

Given a sequence of HS-AFM images capturing each stage in the development of αHL, our goal is to accurately segment the objects of interest and compute their morphological parameters in physical units (nanometers). Formally, let

$$X = \left\{ f_i \in \mathbb{R}^{M\times N} \mid f(x,y) = h, t(f_i) = t(f_i - 1) + \Delta, 0 \le x \le N, 0 \le y \le M \right\} \quad (1)$$

where:

- X is the set of HS-AFM images observed over time,
- M and N represent the image dimensions (length and width),
- $f(x,y)$ denotes the grayscale value at coordinates (x,y), which encodes the height h of the object, and
- $t(f_i)$ records the capture time of the image f_i.

Our objective is to identify and segment the objects of interest in each image. Let

$$P = \left\{ p_j \mid j = 1, \dots, k \right\} \quad (2)$$

be the set of segmented objects (which may represent individual monomers, mushroom-shaped regions, or pores) obtained from an image $x \in X$ using an appropriate segmentation operator S, such that

$$S(x) = P, \qquad (3)$$

Subsequently, for each segmented object $p \in P$, we define a measurement function

$$Q(p) \in \mathbb{R}^d, \quad (4)$$

which computes a vector of morphological parameters such as perimeter, area, height, and volume in physical units (nanometers).

In summary, AFMnanoQ segments the objects of interest whether they are individual monomers, oligomers, mushroom-shaped pre-pores, or complete pores from the HS-AFM images via the function $S(x)$. Once segmented, the subsequent computation of morphological parameters via the function $Q(p)$ allows us to quantitatively analyze and map the evolution of αHL throughout its developmental stages (**Fig. 2**).

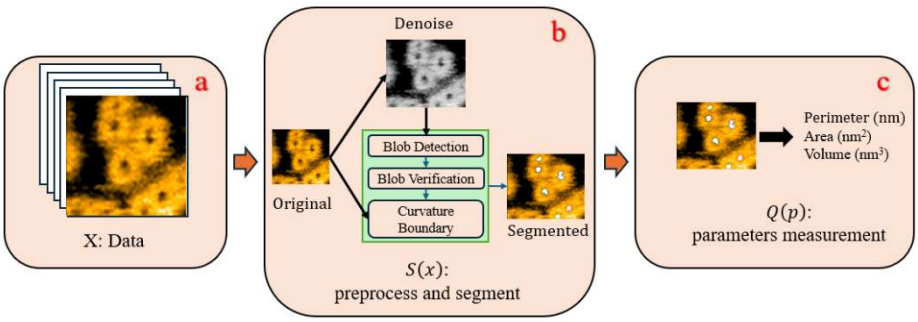


Fig. 2. Overview of the proposed computational framework for analyzing HS-AFM images of αHL. **(a)** the dataset X, a collection of HS-AFM images capturing the dynamic of αHL over time, where each image encodes height information through color-mapped values. **(b)** the image processing function $S(x)$, which encompasses both the preprocessing steps (including denoising and stripe removal) and the segmentation operations that isolate objects of interest (e.g., individual monomers, oligomers, mushroom-shaped regions of pre-pores or pores). Finally, **(c)** depicts the measurement function $Q(p)$, which computes key morphological parameters such as perimeter, area, height, volume, and depth in physical units (nanometers) for each segmented object. Together, these components form a comprehensive pipeline that enables precise quantitative analysis of each developmental stages of αHL as observed in HS-AFM data.

3.1 Denoising

After careful evaluation, a two-pronged approach was adopted to address the primary noise types in AFM images. Non-local Mean[22] filter is applied to mitigate point noise, while the DeStripe[23] method is employed to remove stripe noise. Because stripe noise removal can sometimes result in the loss of fine details and denoising may introduce artifacts, the denoised images are used solely for segmentation. Subsequent steps merge the segmentation masks from the denoised images with the original data, ensuring that morphological parameters are measured accurately while preserving structural fidelity. Within the scope of this paper, traditional algorithms are explored in depth to validate their effectiveness, thereby to establish a foundation for integrating deep-learning techniques to reconstruct large-scale 3D conformations from AFM data (i.e., deepAFMnanoQ-3D).

3.2 Segmentation

Blob Detection: In this initial stage, we identify potential blob-like regions that may represent the cores of objects of interest (**Fig. 3**). A blob is defined as a region in which the pixel intensities differ significantly from the surrounding background and exhibit a relatively homogeneous structure. To detect these regions, we compute the Hessian matrix at each pixel (x, y) of the image I:

$$H(I)(x,y) = \begin{bmatrix} I_{xx}(x,y) & I_{xy}(x,y) \\ I_{xy}(x,y) & I_{yy}(x,y) \end{bmatrix}, \quad (5)$$

where I_{xx} and I_{yy} are the second-order derivatives along the x and y directions, and I_{xy} is the mixed derivative.

The choice of the Hessian matrix, which involves second-order derivatives, is particularly motivated by the characteristics of AFM images. These images typically exhibit smooth gradient variations, meaning that first-order derivative methods (such as Canny or Sobel edge detectors) often fail to detect the subtle boundaries of the objects. In contrast, the second-order derivatives computed by the Hessian matrix are more sensitive to changes in curvature, making them more effective for identifying blob-like structures in this context.

After computing the Hessian, we calculate its determinant $\det\big(H(I)(x,y)\big)$ and identify local maxima as candidate blobs. These regions, characterized by high determinant values, indicate isotropic intensity variations consistent with blob-like structures.

Blob Verification: Building upon the initial detection, we verify candidate blobs by leveraging the characteristic "mushroom" morphology of αHL and the top-down AFM image configuration, which encodes object height in color channels (**Fig. 3**). First, the image is partitioned into multiple height levels using multi-level set clustering; specifically, we apply K-means clustering to establish these levels.

For each height level, a valid candidate blob is expected to be surrounded by a nearly circular shape. In such a shape, the enclosing pixels satisfy the circle equation

$$(x-a)^2 + (y-b)^2 = r^2, \quad (6)$$

where (a,b) represents the candidate blob's center. Considering that blobs have a finite size, we allow the center coordinates to vary within specified bounds, i.e., $x \in [x_{\min}, x_{\max}]$ and $y \in [y_{\min}, y_{\max}]$, and the radius r to vary within $[r_{\min}, r_{\max}]$(**Fig. 3h**).

To determine $r_{\min}$ and $r_{\max}$, raycasting is performed from the blob candidate's center along the eight principal directions. The first intersection with the non-background (objects) is recorded and collected into a set $C_{\min}$. The rays are then extended further until they hit the background again, with those intersection points collected into a set $C_{\max}$. For each set, $C_{\min}$ and $C_{\max}$, the distance from the blob center to each point is calculated. Rays with lengths deviate by more than 1.5 times the median length of the set (either shorter or longer) are discarded. Finally, $r_{\min}$ is defined as the minimum distance between blob center to the remaining points in $C_{\min}$, and $r_{\max}$ as the maximum distance between blob center to the remaining points in $C_{\max}$ (**Fig. 3h**).

Using the points from filtered $C_{\min}$, we then perform region growing to aggregate adjacent pixels that satisfy the defined circle equation, yielding a set X. During this region growing, we partition the area around the blob center into four quadrants and assign each pixel in X to one of these quadrants. We then analyze the distribution of pixels across these quadrants to ensure they are evenly spread, preventing scenarios where the pixels cluster predominantly in one direction despite satisfying the circle equation (**Fig. 3h**).

Finally, we evaluate the connectivity of X through connected component analysis to assess its density and coherence. If X forms a tightly connected region—indicating that the candidate blob is consistently surrounded by a near-circular or arc-shaped boundary across the majority of height levels—then the candidate blob is confirmed as a valid

kernel. This rigorous multi-step verification process ensures that only robust, well-defined blobs are accepted for further analysis (**Fig. 3h**).

Curvature Boundary Extraction: After confirming a blob candidate, the next step is to precisely delineate the object's boundary. Given that AFM images encode 3D structural information as a topographic map, they are well-suited for the Watershed algorithm. In this context, the image is viewed as a landscape, where valleys correspond to homogeneous regions and ridges denote boundaries (**Fig. 3**).

The Watershed algorithm partitions a gradient image G into catchment basins based on local minima. For an image domain Ω with local minima $\{m_i\}$, each catchment basin C_i is defined as:

$$C_i = \{x \in \Omega \mid \forall j \neq i, d_G(x, m_i) < d_G(x, m_j)\}, \quad (7)$$

where $d_G(x, m)$ is a distance metric that incorporates the gradient values along paths from x to m. The watershed lines W are then defined as the set of pixels that lie on the boundaries between these basins:

$$W = \{x \in \Omega \mid x \text{ lies on the boundary between two or more } C_i\}.$$

While the watershed algorithm is prone to over-segmentation in noisy images, we mitigate this issue by restricting its application to the previously verified blob region. This constraint effectively removes extraneous areas from consideration, thereby enhancing the accuracy of the boundary delineation.

In summary, our curvature boundary extraction process combines a modified watershed algorithm—constrained to verified blob regions. This rigorous, multi-stage approach ensures that the object boundaries are accurately and robustly delineated, which is critical for subsequent morphological measurements.

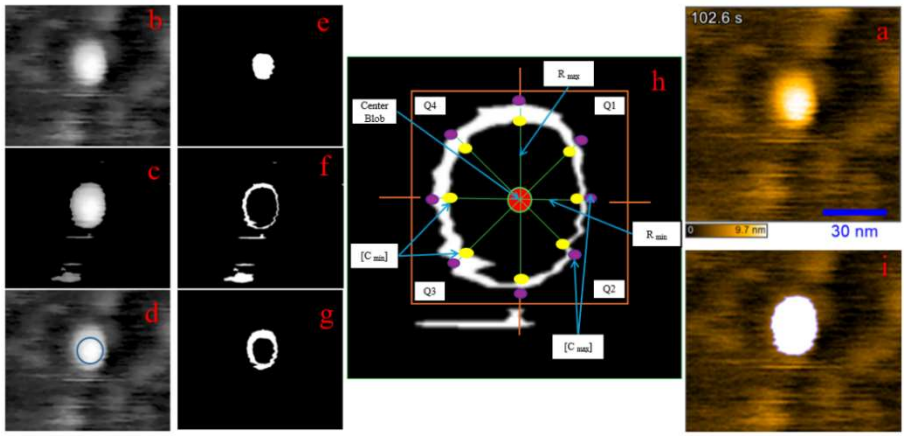


Fig. 3. Segmentation workflow for αHL in the pre-pore stage. **(a)** Original image; **(b)** Grayscale and cropped image (scalebar and elapsed time removed); **(c)** Denoised image with background removed; **(d)** Blob detection identifying potential blobs; **(e,f,g)** Multi-level set clustering based on height; **(h)** Blob verification, "Center Blob" marks potential blob center, [C_{min}] and [C_{max}] denote the initial and extended raycast points used to determine the minimum (R_{min}) and maximum (R_{max}) radius, and Q1, Q2, Q3, Q4 represent quadrants for assessing spatial distribution; **(i)** Curvature boundary extraction precisely delineates and highlights the object boundary.

3.3 Measurement

Once the objects of interest have been accurately segmented, we compute key morphological parameters like the perimeter and the area of each segmented object.

Perimeter Calculation: Let $\{(x_i, y_i)\}_{i=1}^{n}$ denote the set of boundary points extracted from the segmented object, arranged sequentially along its contour. The perimeter P is then computed using a discrete approximation by summing the Euclidean distances between successive points:

$$P = \sum_{i=1}^{n} \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}, \qquad (8)$$

with the convention that $(x_{n+1}, y_{n+1}) = (x_1, y_1)$ to ensure the boundary is closed.

Area Calculation: The area A of the object is determined via integration over the region D enclosed by the boundary. In the continuous domain, this is expressed as:

$$A = \iint_D dx\, dy. \qquad (9)$$

For a discrete image, if the object is represented by a binary mask $M(x, y)$ (where $M(x, y) = 1$ for object, otherwise 0), the area can be approximated by:

$$A \approx \sum_{x,y} M\,(x, y) \cdot \Delta x\, \Delta y, \qquad (10)$$

where Δx and Δy are the physical dimensions (in nanometers) corresponding to each pixel. Alternatively, the area can be computed from the boundary points using Green's theorem [33]:

$$A = \frac{1}{2} \left| \sum_{i=1}^{n} (x_i y_{i+1} - x_{i+1} y_i) \right|, \qquad (11)$$

Conversion to Physical Units After computing the perimeter and area in pixel units, it is necessary to convert these measurements into physical units (nanometers). To achieve this, we exploit the embedded scale bar present in the image (**Fig. 1**). First, we identify the region containing the scale bar by leveraging its known positional and geometric features. Once the scale bar region is isolated, we apply Google Tesseract OCR to extract the textual information indicating the physical length represented by the scale bar.

Let D_{pixel} be the measured length (in pixels) of the scale bar and D_{physical} be the corresponding physical length (in nanometers) as read by OCR. The conversion factor s is then computed as:

$$s = \frac{D_{\text{physical}}}{D_{\text{pixel}}}, \qquad (12)$$

Using this factor, the perimeter P_{pixel} and area A_{pixel} measured in pixels are converted into physical units by:

$$P_{\text{physical}} = s \times P_{\text{pixel}}, \qquad (13)$$

$$A_{\text{physical}} = s^2 \times A_{\text{pixel}}, \qquad (14)$$

This approach ensures that our morphological measurements are accurately translated into nanometer-scale dimensions, which is critical for the quantitative analysis of HS-AFM data.

Volume Calculation: After segmenting the object region and accurately delineating its boundary, we proceed to calculate the volume using the height information encoded in the HS-AFM image. Each pixel (x, y) within the segmented region D is associated with a height value $h(x, y)$. The volume V is defined as the integral of the

difference between a reference baseline height h_0 (typically taken at the boundaries) and the local height over the area D:

$$V = \iint_D [|h_0 - h(x,y)|]\ dx\, dy, \qquad (15)$$

In digital images, this integral is approximated by a discrete sum:

$$V \approx \sum_{(x,y)\in D} [|h_0 - h(x,y)|]\, \Delta x\, \Delta y, \quad (16)$$

where Δx and Δy are the physical dimensions (in nanometers) of each pixel.

Due to limitations in pixel resolution, the raw height data $h(x,y)$ may exhibit abrupt transitions that do not accurately reflect the true continuous surface. To address this, we refine the height information by smoothing the height map. We estimate the local height gradient using finite difference methods to approximate the derivative $\nabla h(x,y)$, which provides an approximation of the rate of change of the height across neighboring pixels. Using this gradient information, a smoothing filter—such as a Gaussian filter—is applied to generate a refined height map $\tilde{h}(x,y)$ that better captures the smooth transition between different height levels.

The volume is then recalculated using the smoothed height values:

$$V \approx \sum_{(x,y)\in D} \left[|h_0 - \tilde{h}(x,y)|\right] \Delta x\, \Delta y, \qquad (17)$$

This two-step process—smoothing the height data to correct for coarse pixel resolution, followed by volume integration—ensures a more accurate estimation of the volume.

After computing the volume in pixel units, we extract the color-to-nanometer conversion factor by reading the color scalebar in image (**Fig. 1**). This factor is combined with the previously obtained pixel-to-nanometer conversion factor, and the volume is converted using the formula:

$$V_{nm} = s_{\text{color}} \times s_{\text{pixel}}^2 \times V, \qquad (18)$$

where $s_{\text{color}} = \frac{\text{color scale value}}{255}$ and s_{pixel} is the pixel-to-nanometer conversion factor.

4 Experiments

4.1 Dataset Description

The dataset comprises HS-AFM image sequences categorized as follows:

- Intermediate Structures: 25 sequences, totaling 6,500 images.
- Pre-Pore Structures: 20 sequences, totaling 4,000 images.
- Pore Structures: 50 sequences, totaling 10,000 images.
- Pre-Pore and Pore Structures: 30 sequences, totaling 9,000 images.

4.2 Denoising Evaluation

We employ the Peak Signal-to-Noise Ratio (PSNR) as the primary metric for assessing the performance of our denoising methods. PSNR is widely used to quantify the quality of reconstructed images by comparing the denoised image with the original noisy image. It is computed based on the mean squared error (MSE) between the two images. The PSNR is defined mathematically as:

$$\text{PSNR} = 10\log_{10}\left(\frac{\text{MAX}_I^2}{\text{MSE}}\right), \qquad (19)$$

where MAX_I represents the maximum possible pixel value in the image (typically 255 for 8-bit images). The MSE is calculated as:

$$\mathrm{MSE} = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}[I(x,y) - I'(x,y)]^2\,, \qquad (20)$$

with $I(x,y)$ denoting the pixel value at position (x,y) in the original image, $I'(x,y)$ the corresponding pixel value in the denoised image, and M and N being the dimensions of the images.

PSNR yields results in decibel (dB) units; a higher PSNR value indicates that the processed image contains less noise. By using the PSNR, we can quantitatively evaluate the effectiveness of our denoising methods in reducing noise while preserving critical image details.

Table 2. PSNR evaluation for denoising methods on original images (include scalebar/text) and cropped images (cropped 8% top, 10% bottom to remove scalebar/text), as shown in **Fig.1**.

Method	Num of Image	PSNR (dB) in original image	PSNR (dB) in cropped image
Median	500	21.26	39.80
Non-local Mean	**500**	**41.51**	**40.98**
DeStripe	500	19.37	25.84
NLM + DeStripe	500	19.40	25.80

Table 2 shows that the Non-local Mean (NLM) method achieves the highest and most stable PSNR. In contrast, methods like DeStripe (and NLM + DeStripe) yield lower PSNRs on original images but perform relatively better after cropping out the scale bar and elapsed text. Based on these results, we primarily adopt NLM for our denoising process, while using DeStripe (recommended in [34]) or its combination with NLM selectively when images are pre-cropped.

4.3 Segmentation Evaluation

To assess the performance of our segmentation framework, we selected a representative subset of the dataset and manually annotated the objects of interest, with a primary focus on images containing pores—considered the most critical for subsequent analyses. The evaluation is conducted in two main categories:

Object Detection Performance: We compare the segmentation results against the manually labeled ground truth by calculating three key metrics: True Positives (TP), which denote the number of correctly detected objects; False Positives (FP), which represent the extra objects erroneously identified; and False Negatives (FN), which indicate the objects that were missed by our framework. The average values of these metrics across the dataset provide a quantitative measure of the detection accuracy of our method **(Table 3)**.

Table 3. Object Detection Evaluation

Method	Correct (TP)	Miss (FN)	Wrong (FP)	Precision TP / (TP+FP)	Recall TP / (TP+FN)
Watershed	6.22	2.13	0.21	0.967	0.745
Blob Detection	7.21	1.18	0.17	0.977	0.859
AFMnanoQ	**7.55**	**0.96**	**0.05**	**0.993**	**0.887**
UNet	7.73	**0.35**	0.48	0.941	**0.957**

Yolov8	**7.91**	0.6	**0.04**	**0.995**	0.929
GroundTruth	8.56	0.0	0.0	1.000	1.000

Additionally, we created a dedicated training dataset (500 samples) in pore stage for deep-learning models by employing a semi-automated labeling approach. Initial annotations were generated using AFMnanoQ, followed by manual refinement to ensure accuracy. This labeled dataset was then used to train UNet and YOLOv8. Strikingly, the results showed that these models achieved high performance on the pore dataset for which they were specifically trained. However, when applied to pre-pore data—samples not included in the training set—their performance dropped significantly, whereas AFMnanoQ remained robust. This highlights the adaptability of our approach, which achieves performance nearly comparable to deep-learning models while maintaining greater generalizability across diverse HS-AFM datasets. The use of AFMnanoQ for training data generation is particularly promising, and in future work, we aim to further explore the application of semi-automated annotation for deep learning models, as well as investigate strategies to enhance model adaptability to heterogeneous nanostructures in AFM and HS-AFM data.

Object Measurement Accuracy: For the accurately detected objects, we further evaluate the segmentation quality in terms of shape and numerical measurements. The shape accuracy is quantified using the overlap ratio (or Intersection over Union, IoU) between the manually annotated objects and the corresponding segmented regions. Additionally, we compare the average values of key morphological parameters—such as perimeter, area, height, and volume—obtained from our framework with reference data extracted from the Protein Data Bank (PDB) and previous studies on αHL. This comprehensive evaluation not only validates the precision of the object detection but also ensures that the measured parameters are in good agreement with established literature values (**Table 4**), based on our feature-driven computational approach.

Table 4. Pore Measurement Evaluation

Method	Number of Sample	Average Perimeter (nm)	Average Area (nm^2)
Manual with UMEX Viewer	1109	11.58 ± 1.14	9.31 ± 1.82
AFMnanoQ	1686	12.68 ± 2.11	10.56 ± 3.58
PDB	-	9.425 ± 5.03	9.08 ± 7.54

Notably, AFMnanoQ measured the height of αHL at approximately 9.94 nm during the pre-pore stage and about 6.46 nm in the pore stage, two states of pre-pores and pores, which were measured from the membrane upward. The results aligns well with the published data[35]. It is important to note that directly applying the volume calculation to images with clearly visible pores is challenging due to tip-sample dilation, noise in pixel and voxel, resolution limitations, and scale conversion issues in AFM data that may lead to inaccurate volume estimates. Therefore, the average pore area and the average height across both pre-pore and pore stages were approximated to infer the inner pore volume. This indirect estimation yielded a pore volume of approximately 70 nm^3, consistent with previously reported values [17],[18],[19].

5 Conclusion

In this study, AFMnanoQ—a feature-driven computational framework—was developed for the segmentation and quantitative analysis of HS-AFM data, in contrast to traditional data-driven approaches. Operating independently of labeled training data, the framework effectively extracts 3D conformational information from 2D surface AFM images, overcoming challenges such as inherent noise and data heterogeneity. Validation on both synthetic and experimental datasets, including detailed analysis of αHL pore formation, demonstrates that AFMnanoQ achieves competitive performance while offering superior robustness and adaptability compared to deep-learning models. Moreover, the high-quality labeled data generated by our framework lays a solid foundation for future deep-learning applications. As the future perspectives, we will use the labeled data produced by AFMnanoQ and integrate computer vision with deep-learning[36] models, aiming to enhance automatic or semi-automatic segmentation accuracy, ultimately bridging the gap between feature-driven and data-driven approaches in AFM and HS-AFM analysis. This serves as the first step for developing deepAFMnanoQ-3D to generate large-scale 3D conformations from AFM data.

Acknowledgments. This research is supported by research funding from Faculty of Information Technology, University of Science, Viet Nam National University – Ho Chi Minh City.

References

1. Ando, T., Fukuda, S., Ngo, K. X. & Flechsig, H. High-Speed Atomic Force Microscopy for Filming Protein Molecules in Dynamic Action. Annu. Rev. Biophys. 53, 19–40 (2024).
2. Heath, G. R. et al. Localization atomic force microscopy. Nature 594, 385–390 (2021).
3. Heath, G. R., Micklethwaite, E. & Storer, T. NanoLocz: Image analysis platform for AFM, high-speed AFM and localization AFM. Small Meth. 2301766, 1–9 (2024).
4. Niina, T., Matsunaga, Y. & Takada, S. Rigid-body fitting to atomic force microscopy images for inferring probe shape and biomolecular structure. PLoS Comput. Biol. 17, 1–24 (2021).
5. Amyot, R. & Flechsig, H. BioAFMviewer: An interactive interface for simulated AFM scanning of biomolecular structures and dynamics. PLoS Comput. Biol. 16, 1–12 (2020).
6. Amyot R., Marchesi A., Franz C. M., Casuso I. & Flechsig H: Simulation atomic force microscopy for atomic reconstruction of biomolecular structures from resolution-limited experimental images. PLoS Comput. Biol.18,1–15 (2022).
7. Carracedo-Cosme, J., Romero-Muñiz, C., Pou, P. & Pérez, R. QUAM-AFM: A Free Database for Molecular Identification by Atomic Force Microscopy. J. Chem. Inf. Model. 62, 1214–1223 (2022).
8. Carracedo-Cosme, J. & Pérez, R. Molecular identification with atomic force microscopy and conditional generative adversarial networks. npj Comput. Mater. 10, (2024).
9. Vuillemot, R., Pellequer, J. & Grudinin, S. AFMfit: Deciphering conformational dynamics in AFM data using fast nonlinear NMA and FFT-based search. bioRxiv (2024). doi:10.1101/2024.06.03.597083
10. Hoffmann A., Grudinin S. NOLB: Nonlinear Rigid Block Normal-Mode Analysis Method. J. Chem. Theory Comput. 13, 2123–2134 (2017).
11. Niina T., Fuchigami S. & Takada S. Flexible Fitting of Biomolecular Structures to Atomic Force Microscopy Images via Biased Molecular Simulations. J. Chem. Theory Comput. 16, 1349–1358 (2020).
12. Jiang, Y., Wang, Z. & Scheuring, S. A structural biology compatible file format for atomic force microscopy. Nat. Commun. 16, 1–16 (2025).

13. Tunyasuvunakool, K. et al. Highly accurate protein structure prediction for the human proteome. Nature 596, 590–596 (2021).
14. Abramson, J. et al. Accurate structure prediction of biomolecular interactions with AlphaFold 3. Nature 630, 493–500 (2024).
15. Degenhardt, M. F. S. et al. Determining structures of RNA conformers using AFM and deep neural networks. Nature 637, (2025).
16. Lane, T. J. Protein structure prediction has reached the single-structure frontier. Nat. Methods 20, 170–173 (2023).
17. Daniel M Czajkowsky, Sitong Sheng, Zhifeng Shao: Staphylococcal alpha-hemolysin can form hexamers in phospholipid bilayers. Journal of Molecular Biology, 276(2), 325-330. https://doi.org/10.1006/jmbi.1997.1535 (1998)
18. Sugawara T, Yamashita D, Kato K, Peng Z, Ueda J, Kaneko J, Kamio Y, Tanaka Y, Yao M.: Structural basis for pore-forming mechanism of staphylococcal α-hemolysin. Toxicon. 2015 Dec 15;108:226-31. doi: 10.1016/j.toxicon.2015.09.033. Epub 2015 Sep 30. PMID: 26428390. (2015)
19. Song L, Hobaugh MR, Shustak C, Cheley S, Bayley H, Gouaux JE.: Structure of staphylococcal alpha-hemolysin, a heptameric transmembrane pore. Science. 1996 Dec 13;274(5294):1859-66. doi: 10.1126/science.274.5294.1859. PMID: 8943190. (1996)
20. Adhikari RP, Karauzum H, Sarwar J, Abaandou L, Mahmoudieh M, Boroun AR, Vu H, Nguyen T, Devi VS, Shulenin S, Warfield KL, Aman MJ.: Novel structurally designed vaccine for S. aureus α-hemolysin: protection against bacteremia and pneumonia. PLoS One. 2012;7(6):e38567. doi: 10.1371/journal.pone.0038567. Epub 2012 Jun 6. PMID: 22701668; PMCID: PMC3368876. (2012)
21. Fechner, P. et al. Structural information, resolution, and noise in high-resolution atomic force microscopy topographs. Biophys. J. 96, 3822–3831 (2009).
22. Antoni Buades, Bartomeu Coll, Jean-Michel Morel: Non-Local Means Denoising. Image Processing On Line, pp. 208–212. https://doi.org/10.5201/ipol.2011.bcm_nlm (2011)
23. Chen, Sw.W., Pellequer, JL. DeStripe: frequency-based algorithm for removing stripe noises from AFM images. BMC Struct Biol 11, 7. https://doi.org/10.1186/1472-6807-11-7 (2011)
24. Rafael C. Gonzalez, Richard E. Woods: Digital image processing 3rd Edition (2008)
25. G. Chen, W. Xie and Y. Zhao, "Wavelet-based denoising: A brief review," 2013 Fourth International Conference on Intelligent Control and Information Processing (ICICIP), Beijing, China, 2013, pp. 570-574, doi: 10.1109/ICICIP.2013.6568140. (2013)
26. Li, M., Rieck, J., Noheda, B. et al.: Stripe noise removal in conductive atomic force microscopy. Sci Rep 14, 3931. https://doi.org/10.1038/s41598-024-54094-w (2024)
27. Hoichan Jung, Giwoong Han, Seong Jun Jung, Sung Won Han: Comparative study of deep learning algorithms for atomic force microscopy image denoising. Micron, Volume 161,103332,ISSN 0968-4328 (2022)
28. Marsh BP, Chada N, Sanganna Gari RR, Sigdel KP, King GM. The Hessian Blob Algorithm: Precise Particle Detection in Atomic Force Microscopy Imagery. doi: 10.1038/s41598-018-19379-x (2018)
29. V. Badrinarayanan, A. Kendall and R. Cipolla: "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 12, pp. 2481-2495, 1 Dec. 2017, doi: 10.1109/TPAMI.2016.2644615. (2017)
30. Ronneberger, O., Fischer, P., Brox, T: U-Net: Convolutional Networks for Biomedical Image Segmentation. In: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. MICCAI 2015. Lecture Notes in Computer Science, vol 9351. Springer, Cham. https://doi.org/10.1007/978-3-319-24574-4_28 (2015)

31. Sohan, M., Sai Ram, T., Rami Reddy, C.V.: A Review on YOLOv8 and Its Advancements. In: Data Intelligence and Cognitive Informatics. ICDICI 2023. Algorithms for Intelligent Systems. Springer, Singapore. https://doi.org/10.1007/978-981-99-7962-2_39 (2024)
32. Ngan Thi Phuong Le el al. Rencontres du Vietnam: The 1st & 2nd International Symposium of Nano Life Science: Nano Biotechnology, Biophysics, Computation (NanoBioCoM2023, NanoBioCoM2024). 2023, 2024
33. Lang, S.: Green's Theorem. In: Calculus of Several Variables. Undergraduate Texts in Mathematics. Springer. https://doi.org/10.1007/978-1-4612-1068-9_10 (1987)
34. Duy Nguyen Thanh Phan, Hung Nguyen Long Tran: Developing an HS-AFM Image Analysis System for Protein Pores Formed by Toxins. Bachelor's Thesis supervised by Ngoc Quoc Ly. University of Science – VNU-HCM (2024)
35. Song, L. et al. Structure of staphylococcal α-hemolysin, a heptameric transmembrane pore. Science (80-.). 274, 1859–1866 (1996).
36. Paruchuri, A. et al. Machine Learning for Analyzing Atomic Force Microscopy (AFM) Images Generated from Polymer Blends. Digital Discovery, 3, 2533-2550 (2024).