AFM-Fold: Rapid Reconstruction of Protein Conformations from AFM Images

Tsuyoshi Kawai[1] and Yasuhiro Matsunaga*[1,2]

[1]Graduate School of Science and Engineering, Saitama University
[2]RIKEN Center for Computational Science

November 17, 2025

Abstract

High-speed atomic force microscopy (HS-AFM) enables direct visualization of protein dynamics under near-physiological conditions, yet its intrinsic limitation to surface topography prevents atomic-level structural characterization. We present AFM-Fold, a generative AI-based framework that reconstructs three-dimensional protein conformations directly from AFM images. AFM-Fold combines a rotation-equivariant convolutional neural network, which extracts low-dimensional collective variables (CVs) from AFM images, with a guided sampling strategy that steers the diffusion process of AlphaFold 3 toward conformations consistent with the inferred CVs. Using pseudo-AFM images of adenylate kinase, AFM-Fold accurately reproduced both open and closed conformations, achieving sub-Å reconstruction errors. Application to 159 experimental HS-AFM frames of the flagellar protein $\mathrm{FlhA_C}$ further demonstrated that AFM-Fold outperforms rigid-body fitting and captures time-correlated domain motions that reflect underlying conformational dynamics. AFM-Fold enables rapid, physically plausible structure estimation from individual AFM images, typically within one minute per frame, without relying on molecular dynamics simulations. This unified and computationally efficient pipeline opens a route to high-throughput structural analysis of HS-AFM movies.

1 Introduction

High-speed atomic force microscopy (HS-AFM) enables direct visualization of biomolecular dynamics in solution under near-physiological conditions, providing unprecedented insights into the relationship between biomolecular conformational dynamics and biological function [1, 2]. However, AFM measurements are inherently limited to surface topography, and the achievable spatial resolution is fundamentally constrained by the finite size of the cantilever tip [3, 4]. To address these limitations, there is a compelling need for computational modeling approaches that can integrate experimental AFM data with atomic-resolution three-dimensional structural models.

One approach is rigid-body fitting, in which template structures are selected from structural databases or molecular dynamics (MD) simulations and then exhaustively rotated and translated to find the best pose matching the AFM image [5–13]. Although powerful, this approach cannot account for conformational flexibility or changes; consequently, rigid-body fitting often fails to provide accurate structural models when the template library lacks conformations that match the AFM image. In real biological systems, the function of biomolecules is often coupled with conformational flexibility or dynamics. Thus, methods accounting for such conformational flexibility are required for analyzing and interpreting experimental AFM data.

In recent years, methods have been proposed that utilize MD or Monte Carlo (MC) simulation to generate conformations consistent with AFM images. For example, in flexible fitting [14], the agreement with an AFM image is defined as an additional potential energy function V_{AFM}, and MD simulations are driven to minimize it. In flexible fitting, sampling is often trapped in local minima, making

*ymatsunaga@riken.jp

optimization slow when separated by high barriers. If the global minimum is desired, prohibitively long simulations must be performed to ensure adequate sampling of the conformational space. To reduce the computational cost of MD, NMFF-AFM [15, 16] uses a subset of normal modes to find conformations consistent with AFM images. In this case, however, the conformational space is fundamentally limited to capturing only harmonic fluctuations around a single reference structure. Another approach is to use a 3D Gaussian mixture model combined with MC sampling to estimate 3D low-resolution protein structures with conformational flexibility [17, 18].

In the field of AI-based structure modeling or prediction, sequence-to-structure models such as AlphaFold 2 [19–21], AlphaFold 3, and Boltz-1 [22] have demonstrated unprecedented precision in predicting the 3D fold of a protein. These models can be regarded as generative AI models that generate atomistic conformational ensembles from a given sequence. In principle, these generative AI models can generate not only specific peak structures present in the training data, but also diverse structural conformations. Moreover, it is known that the generation process can be navigated by applying conditioning or external forces during the generation stage. In the field of image generation, research has been conducted on navigating the generation process to bring generated images closer to specific targets. Recently, guiding the generative process of diffusion-based structure prediction models, including AlphaFold 3 and Boltz-1, has emerged as an approach aimed at generating targeted conformational ensembles. For example, in the context of NMR and X-ray crystallography data analysis [23], it has been demonstrated that imposing guided restraints during the generative process enables the generation of ensembles consistent with experimental observations. Similarly, for cryo-EM data [24], the incorporation of global or local density-map fitting as restraints has been proposed as a means of navigating the predicted conformations toward agreement with the experimental data.

In this study, we introduce AFM-Fold, which extends these ideas to AFM data analysis by employing structural restraints extracted from AFM images to guide the sampling trajectories of pre-trained protein structure generative models. Specifically, in the first step, a group-invariant convolutional neural network (CNN) [25] estimates low-dimensional collective variables (CVs, such as inter-domain distances) from AFM images, and then these estimated values are used as restraints to guide the generative process, enabling the rapid sampling of 3D conformations consistent with AFM images. We validated AFM-Fold in twin experiments. First, using adenylate kinase (AK, PDB IDs: 1AKE [26] and 4AKE [27]), we estimated the underlying CV values and accurately reconstructed conformations close to the ground truth. Second, we applied AFM-Fold to real experimental HS-AFM data from the flagellar protein $FlhA_C$ (PDBID: 3A5I [28]) and confirmed that it predicts conformations more consistent with AFM images than the crystal structure fitted by rigid-body fitting calculations. Furthermore, we confirmed that AFM-Fold captures the time-correlated domain motions that reflect underlying conformational dynamics.

A distinctive feature of our framework, AFM-Fold, is that all steps—from training data generation, model training, and inference, to structure generation—are integrated within a single unified pipeline, without relying on time-consuming processes such as molecular dynamics simulations. This enables rapid reconstruction of structures from AFM images while reducing computational cost, allowing the observation of structural dynamics from hundreds of consecutive HS-AFM images. AFM-Fold provides a powerful framework for observing conformational transitions and functional mechanisms underlying HS-AFM images.

2 Related Work

Generative AI-based approaches employing diffusion models or flow models enable us to efficiently sample molecular conformations in a multiscale manner. For example, recent structure-generation foundation models, including AlphaFold 3 and Boltz-1, employ diffusion-based formulations. In particular, Boltz-2 [29] introduced *boltz-steering*, a framework that imposes a customized potential to navigate the sampling towards physically plausible conformations. These models largely follow a "one sequence–one structure" paradigm, which does not focus on producing diverse conformational ensembles or capturing conformational flexibility.

Several approaches have been proposed to address this problem. For example, the MSA subsampling technique [30] and its extensions [31–33] increase the diversity of multiple sequence alignments (MSAs) by masking parts of the MSA. Another powerful direction is the development of ensemble generative or MD surrogate models. An early contribution in this area is the Boltzmann generator

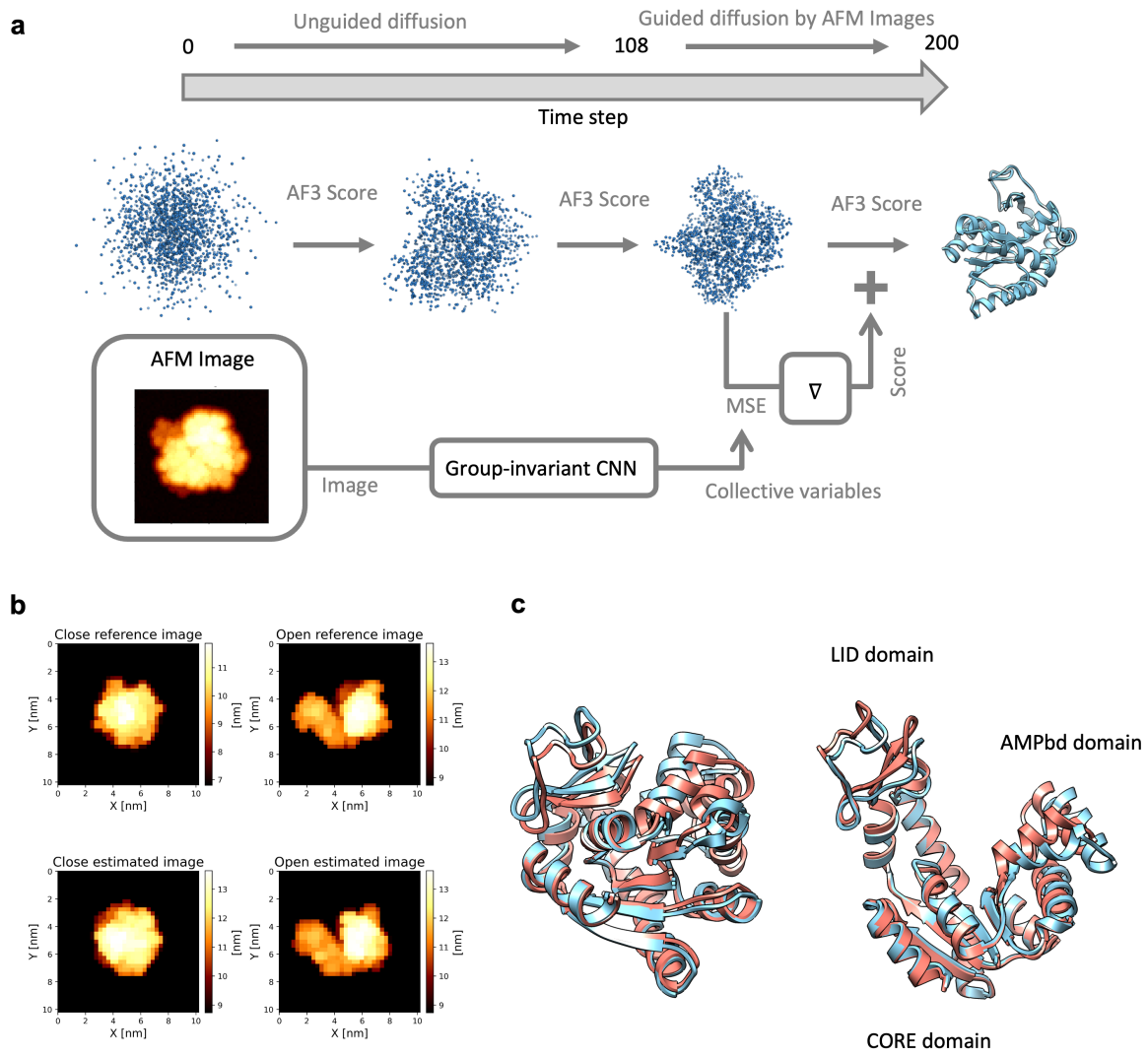

Figure 1: **Schematic overview of AFM-Fold and outcome**. AFM-based guidance navigates AlphaFold 3 and reproduces the closed and open states of AK. **(a)** Schematic diagram of AFM-Fold. The AFM image, shown at the lower left, serves as the input and is processed as follows: (1) coordinates in the inter-domain distance space are predicted from the AFM image by a group-invariant CNN (2) during the generative process, where the diffusion step i progresses discretely from 0 to 200, the mean squared error (MSE) between the predicted coordinates and the generated conformations is computed when $i > 108$; (3) the gradient of the MSE is used as a correction score to guide the generative process of AlphaFold 3. **(b)(c)** Results of structure prediction from pseudo-AFM images of AK. In (b), the upper panels show the reference pseudo-AFM images generated from the conformations in the PDB database (close: 1AKE, open: 4AKE). The lower panels show the reproduced pseudo-AFM images generated from the structures estimated by AFM-Fold. In (c), the blue structures (left: closed, right: open) represent the reference conformations, which were used to generate upper panels in (b). The red structures are the corresponding structures estimated by AFM-Fold.

[34], which was designed to learn and reproduce the Boltzmann distribution of small, specific peptides. More recently, models have been developed to generate diverse structural ensembles from sequences by learning from extensive MD simulation data across a wide range of proteins [35–41]. In these methods, obtaining sufficiently long MD trajectories for training requires substantial computational resources, and such models inevitably inherit the force-field biases present in the training data.

To mitigate this issue, recent studies have explored conditioning ensemble generative models on experimental measurements or physical prior knowledge. Examples include methods that generate NMR-consistent structural ensembles [23, 42, 43], and cryo-EM-consistent modeling using Boltz-1 and Chroma [24, 44]. Following this direction, AFM-Fold enables AFM-conditioned conformation generation, guiding AlphaFold 3's prior distribution toward modeling more dynamic conformational states captured in HS-AFM images.

3 Methods

Our AFM-Fold framework leverages AlphaFold 3 (here, we used the open-sourced Protenix model [45], a PyTorch reimplementation of AlphaFold 3) to estimate the underlying 3D conformation from a given AFM image. To generate 3D structures consistent with AFM images using AlphaFold 3, we follow two main steps: (i) first, we estimate low-dimensional collective variables (CVs) for the target molecule from the given AFM image using a pre-trained CNN. For CVs, we use inter-domain distances, assuming multi-domain proteins as a typical case. (ii) Next, we use the estimated CVs to add restraints to AlphaFold 3 during structure generation, producing 3D structures that are consistent with the AFM image (see Fig. 1a).

In this section, we present our methodology in three parts. In Section 3.1, we summarize the group-equivariant CNN used to estimate the CVs from AFM images. In Section 3.2, we describe how we add restraints in the diffusion process of AlphaFold 3 using the estimated CVs. Finally in Section 3.3, the overall computational protocol of our AFM-Fold framework is given.

3.1 Group-equivariant CNNs

In AFM 2D image analysis, similar to single particle analysis of cryo-EM micrographs, the pose of the 3D structure is unknown. The alignment calculation between a 2D image and the 3D structural pose requires an exhaustive search and is computationally expensive. Moreover, if misalignment occurs, it leads to extreme deterioration in estimation accuracy. In the case of cryo-EM, the influence of misalignment could be mitigated on average by a vast number of micrographs, but in the case of HS-AFM, since we want to perform structural estimation for each single image to infer underlying conformational dynamics, misalignment cannot be tolerated. Therefore, we use a rotation-equivariant CNN model (*group-equivariant* CNN; g-CNN) [25, 46–48] that directly estimates structure-related CVs from 2D AFM images without performing alignment calculations.

In navigating the structure generation process of AlphaFold 3 to generate 3D structures consistent with AFM images, it would be possible to optimize by calculating image similarity at each step, as done in rigid-body fitting. In this case, higher-resolution image reproduction is expected, but on the other hand, we empirically found that it tends to produce structures trapped in local minima. This behavior arises because the image similarity landscape is highly multimodal and extremely sensitive to translational and in-plane rotational differences. Under a feasible amount of alignment search, inaccurate gradients are continuously applied, resulting in misleading forces during optimization. Therefore, instead of relying on image-to-image correlation, it was necessary to construct a comparison measure that is robust to in-plane rotations and translations of AFM images. In particular, to handle a large number of AFM frames, the similarity between the generated intermediate structures and the images needed to be computed efficiently.

With this motivation in mind, we summarize the theory of g-CNN [25, 46–48], and describe how it enables efficient estimation of CVs from AFM images. A standard CNN is naturally *translation equivariant* but not rotation equivariant. In contrast, g-CNN modifies feature indexing and kernel weight sharing so that the entire network becomes equivariant with respect to a chosen rotation group. Utilizing this architecture, the same physical state yields an identical representation regardless of its placement on the image plane, thereby reducing the need for extensive data augmentation and

shortening training time. Consequently, we employ the g-CNN to estimate CVs from a single AFM image.

We model an AFM image as a single-channel (scalar) field $I : \mathbb{R}^2 \to \mathbb{R}$ with pixel coordinate $x = (x_1, x_2) \in \mathbb{R}^2$. In a standard CNN, intermediate feature maps are represented as functions $f : \mathbb{R}^2 \to \mathbb{R}^c$. The input image is first mapped to a c-channel field by convolution with channel-specific filters $\{\psi_i\}_{i=1}^c$:

$$(\psi * I)(x)[i] := \int_{\mathbb{R}^2} \psi_i(y)\, I(x-y)\, dy, \tag{1}$$

where $*$ is convolution and dy denotes the area element. In practice, the integral is implemented as a finite sum over pixels; we keep the continuous notation for clarity.

Here, we consider the planar motion group $\mathbb{R}^2 \rtimes C_N$, where $C_N = \{e, r, \ldots, r^{N-1}\} \subset \mathrm{SO}(2)$ is a finite rotation subgroup. For scalar or standard c-channel fields (without group indexing that we later explain), the action $\pi(t, g)$ of translation $t \in \mathbb{R}^2$ and rotation $g \in C_N$ for the field u is $(\pi(t, g)u)(x) := u\big(g^{-1}(x - t)\big)$.

Because convolution commutes with translations, a standard CNN is translation-equivariant. For a single convolution with a (matrix-valued) kernel $k(y) \in \mathbb{R}^{c_{\text{out}} \times c_{\text{in}}}$,

$$\begin{aligned} \big(k * (\pi(t, e)u)\big)(x) &= \int_{\mathbb{R}^2} k(y)\, u(x - y - t)\, dy \\ &= (k * u)(x - t) = \big(\pi(t, e)(k * u)\big)(x). \end{aligned} \tag{2}$$

Since a typical activation σ is pointwise, σ is also translation-equivariant, $\sigma(\pi(t, e)u) = \pi(t, e)\sigma(u)$ and the property propagates through the network. By contrast, the standard convolution is generally *not* rotation-equivariant.

The key idea of g-CNNs is to *lift* an image to a feature field indexed by group elements and to impose weight sharing consistent with the group action. Given a base filter $\psi : \mathbb{R}^2 \to \mathbb{R}$, define its rotated copies $\psi_h(y) := \psi(h^{-1}y)$ for $h \in C_N$, and set

$$v(x)[h] \;:=\; \int_{\mathbb{R}^2} \psi(h^{-1}y)\, I(x - y)\, dy, \qquad h \in C_N. \tag{3}$$

Thus $v : \mathbb{R}^2 \times C_N \to \mathbb{R}$ is a group-indexed (orientation-channel) field.

For such lifted fields, if the kernel satisfies the following constraint about sharing weights, $k(gy)[h, h'] = k(y)\big[g^{-1}h, g^{-1}h'\big]$ for all $g, h, h' \in C_N$, then the group convolution commutes with $\pi(0, g)$:

$$\big(k * (\pi(0, g)v)\big)(x)[h] \;=\; \big(\pi(0, g)(k * v)\big)(x)[h]. \tag{4}$$

The proof can be found in Appendix D.1 of [25]. Hence, adding to the translation equivariance Eq. (2), we can construct CNNs equivariant to the chosen discrete rotation group C_N. Channel-wise activations commute with the orientation relabeling, so equivariance to rotations (and translations) is preserved layer by layer.

A practical architecture uses multiple lifted blocks $v_1, \ldots, v_b$; the overall action is the block-diagonal (direct-sum) representation $\pi = \bigoplus_{i=1}^b \pi_i$. In AFM-Fold, because we require a vector invariant to rotations and translations, we apply group pooling and spatial pooling per block at last; i.e.,

$$\mathrm{GroupPooling}(v_i)(x) := \mathrm{Max}_{h \in C_N} v_i(x)[h], \quad \mathrm{SpatialPooling}(v_i) := \mathrm{Mean}_{x \in \mathbb{R}^2} v_i(x), \tag{5}$$

which yields features invariant to translations and rotations.

3.2 Navigating AlphaFold 3 with inter-domain distance restraints

Next, we describe a method for imposing a restraint on the generative process of AlphaFold 3 so that the CVs approach user-specified target values. Specifically, while AlphaFold 3 employs an Elucidated Diffusion Model (EDM) formulation [49], we compute the likelihood of the generated structure with respect to the specified CVs and add it to the original AlphaFold 3 score as a driving force that increases this likelihood. With this modified score, the generated structures become explicitly conditioned on the CVs.

Let a denote an amino acid sequence and $X = (x_1, \ldots, x_m)$ the 3D coordinates of all atoms, where $x_i \in \mathbb{R}^3$. We write X_{t_i} for the state at diffusion time t_i ($T > t_1 > t_2 > \cdots > t_N > 0$). Sampling the generative process of AlphaFold 3 can be written as the stochastic EDM sampler:

$$\begin{aligned} X_{t_{i+1}} &= X_{t_i} - \eta(t_{i+1} - \hat{t}_i)\nabla_{X_{t_i}} \log p_{t_i}(X_{t_i}|a) + \lambda\sqrt{\hat{t}_i^2 - t_i^2}\epsilon, \\ \nabla_{X_{t_i}} \log p_{t_i}(X_{t_i}|a) &\approx -\big(X_{t_i} - \mathrm{Denoiser}(X_{t_i}|a)\big)/\hat{t}_i, \end{aligned} \tag{6}$$

Here, $\hat{t}_i$ denotes a time larger than t_i, defined as $\hat{t}_i = (1 + \gamma_i)t_i$, with a parameter γ_i that is set to 0.8 when $t_{i+1} > 1.0$, and 0 otherwise. The coefficients η and λ are hyperparameters called the *noise scale* and *step scale*, respectively, and are set to $\eta = 1.5$ and $\lambda = 1.003$. $\mathrm{Denoiser}(\cdot|a)$ represents a denoiser conditioned by coevolutionary information about a.

Let $\phi_{\mathrm{target}} \in \mathbb{R}^D$ denote the estimated coordinate in the CV space, where D is the dimension of the CVs. By Bayes' rule, the conditional score decomposes as

$$\nabla_{X_t} \log p_t\big(X_t \big| \phi_{\mathrm{target}}, a\big) = \nabla_{X_t} \log p_t(X_t|a) + \nabla_{X_t} \log p_t(\phi_{\mathrm{target}}|X_t, a). \tag{7}$$

Therefore, we can bias the stochastic sampler by adding the guidance term:

$$X_{t_{i+1}} = X_{t_i} - \eta(t_{i+1} - \hat{t}_i)\big(\nabla_{X_{t_i}} \log p_{t_i}(X_{t_i}|a) + \nabla_{X_{t_i}} \log p_{t_i}(\phi_{\mathrm{target}}|X_{t_i}, a)\big) + \lambda\sqrt{\hat{t}_i^2 - t_i^2}\epsilon \tag{8}$$

so that the process is guided toward $\phi(X_{t_N}) \approx \phi_{\mathrm{target}}$, where $\phi : \mathbb{R}^{3m} \to \mathbb{R}^D$ is a differentiable mapping from structures to this feature space.

In this work, we replace the guidance score by the gradient of a squared-error loss:

$$\mathcal{L}_{\mathrm{MSE}}(X) \;=\; \big\|\phi(X) - \phi_{\mathrm{target}}\big\|_2^2, \tag{9}$$

$$\nabla_{X_t} \log p_t(\phi_{\mathrm{target}}|X_t, a) \approx -\eta_t \nabla_{X_t} \mathcal{L}_{\mathrm{MSE}}(X_t), \tag{10}$$

which is consistent with an isotropic Gaussian model $p_t\big(\phi_{\mathrm{object}}|X_t, a\big) \propto \exp\big(-\|\phi(X_t) - \phi_{\mathrm{object}}\|_2^2/(2\sigma_t^2)\big)$, in which case $\eta_t = 1/(2\sigma_t^2)$. The optimal selection of η_t is analyzed in SI section S1.

3.3 AFM-Fold computational protocol

Our AFM-Fold framework consists of two main steps as described in the previous subsections: (i) estimation of CVs from AFM images using a g-CNN, and (ii) navigating the generative process of AlphaFold 3 using the estimated CVs. However, additional steps are still required for preparing training data for the g-CNN, training, inference, and evaluation. In this subsection, we describe these steps.

3.3.1 Preparation of training data

To train the g-CNN to learn the relationship between AFM images and underlying 3D structures, we construct training data by generating pseudo-AFM images from 3D conformations and pairing them with ground-truth CV labels. In this study, these conformations are generated by applying diverse CV-based restraints to AlphaFold 3. This strategy offers two key advantages: (i) first, compared to other computational tools to generate 3D structures (such as MD simulations), the required computational resources are substantially reduced; (ii) second, this approach allows for sufficient sampling of intermediate conformations so that the g-CNN can learn to interpolate the structures in the conformational space. In order to detect smooth conformational transitions, the g-CNN requires training data not only for two boundary conformations (e.g., start and end conformations) but also for intermediate conformations continuously linking them. In the absence of such intermediates, the g-CNN is unable to accurately model continuous structural variability. By specifying the CV values continuously, we can generate diverse conformations that cover the conformational space.

Generating candidate conformations to broadly cover the CV space We sampled diverse conformations by AlphaFold 3, broadly covering the CV space specifying the CV values with grids. To do this, first, we obtain a reference structure X_{ref} with AlphaFold 3 with no restraints. Then, target CV vectors $\{\phi^i_{\text{perturb}}\}$ are placed on a grid around $\phi(X_{\text{ref}})$ such that, for each dimension i,

$$\phi^i_{\text{perturb},d} \in \left[\min(0.0\text{ nm}, \phi^i(X_{\text{ref}}) - 5.0\text{ nm}), \phi^i(X_{\text{ref}}) + 5.0\text{ nm}\right], \qquad \forall i. \tag{11}$$

The grid used a step of 0.5 nm for each axis. Then, the generation with AlphaFold 3 is conducted using each $\{\phi^i_{\text{perturb}}\}$ as the restraints for the navigating process.

Geometric sanitization The candidate conformations generated in the previous step may contain geometric violations (e.g., steric clashes, backbone/side-chain outliers). Such conformations could bias the g-CNN by introducing unphysical patterns without valid conformational counterparts. To exclude such data from the training set, we applied thresholds to perform geometric sanitization, based on MolProbity scores (Table 1), excluding clear violations. This procedure constrains the g-CNN to perform inference within a conformational space devoid of geometric violations. The resulting set that passes this sanitization serves as a training set for training the g-CNN and for synthesizing pseudo-AFM images used in training.

Table 1: MolProbity score thresholds used for geometric sanitization.

Criterion	Threshold
MolProbity Score	≤ 4.0
Clash Score	≤ 70.0
Ramachandran favored (%)	≥ 90.0
Rotamer Outlier (%)	≤ 50.0

Pseudo-AFM image rendering. In rendering the pseudo-AFM, we used the morphology functions implemented in our previous work [4]. The specific parameter settings are summarized in Table 2. It should be noted that for the test case of AK, we used smaller values for the tip radius and finer resolution of pseudo-AFM images compared to typical experimental conditions. On the other hand, for FlhA_C, since the precise tip geometry is unknown in experiments, the probe radius r was sampled uniformly within a range. Moreover, the height distribution of pseudo-AFM images was fitted to the real experimental images using `skimage.exposure.match_histograms`. For both AK and FlhA_C, we used the inter-domain distances (distances between the centers of mass of Cα atoms in the domains) as the CVs.

Table 2: Settings for training pseudo-AFM images.

	AK	FlhA_C
Lateral resolution [nm/pixel]	0.3	0.98
Pixel size [pixel × pixel]	35×35	35×35
Tip radius [nm]	$r \sim \text{Uniform}(0.3, 0.6)$	$r \sim \text{Uniform}(2.0, 6.0)$
Tip apex angle [degree]	$a \sim \text{Uniform}(10, 30)$	$a \sim \text{Uniform}(10, 30)$
Noise std. dev. [nm]	0.0	0.5
Histogram matching	—	✓
Dataset size [frames]	5M	5M
Collective variables	Inter-domain distances (LID-CORE, CORE-AMPbd, AMPbd-LID)	Inter-domain distances ($\text{A}_\text{C}\text{D}_1$-$\text{A}_\text{C}\text{D}_4$, $\text{A}_\text{C}\text{D}_2$-$\text{A}_\text{C}\text{D}_3$, $\text{A}_\text{C}\text{D}_2$-$\text{A}_\text{C}\text{D}_4$)

3.3.2 Training

Here, the training of the g-CNN can be regarded as supervised regression, where the task is to predict the CV values from a given AFM image. As the most straightforward loss function for supervised regression, we adopted the mean squared error (MSE) loss in the CV space:

$$\mathcal{L}_{\text{CNN}} = \text{Mean}_i \|\phi_{\text{pred}}(I_i) - \phi_{\text{true}}(I_i)\|^2, \tag{12}$$

where I_i denotes the i-th input pseudo-AFM image, $\phi_{\text{pred}}(I_i)$ is the CV values predicted by the g-CNN, and $\phi_{\text{true}}(I_i)$ is the corresponding ground-truth CV values. As the g-CNN architecture, we implemented a $(\mathbb{R}^2, +) \rtimes C_8$-invariant CNN (see SI section S2 for the implementation details).

All models in this paper were trained on a single node equipped with an NVIDIA RTX A6000 GPU (48 GB memory). The training took approximately 2–3 days for both AK and FlhA_C.

3.3.3 Inference

The inference step consists of the estimation of CVs (inter-domain distances) from the AFM image by the g-CNN, and the navigation of the generative process of AlphaFold 3 using the estimated CVs. In this study, we require that the MSE of the g-CNN for the generated structures be below sub-Å, which is an important resolution for distinguishing different conformations for both AK and FlhA_C.

The computation time for the inference step was extremely short compared to flexible fitting calculations using MD simulations. Indeed, inference time was less than one minute per image for both AK and FlhA_C.

3.3.4 Evaluation

To evaluate the accuracy of the estimated conformation, we computed the root mean square deviation (RMSD) between the estimated and the ground-truth conformations. Also, to evaluate how well the estimated conformation reproduces the given AFM image, we performed a rigid-body fitting, optimizing 3D rotation, translation, and tip radius. The optimization is evaluated through the correlation coefficient (c.c.) with the given AFM image, using Eq. (13) [14]. The pose achieving the maximum c.c. value across all rotations, translations and tip radii was taken as the optimal pose.

$$\text{c.c.}(R) = \frac{\sum_{p \in \text{pixels}} H_p^{(\text{exp})} H_p^{(\text{sim})}(R)}{\sqrt{\sum_{p \in \text{pixels}} \left(H_p^{(\text{exp})}\right)^2} \sqrt{\sum_{p \in \text{pixels}} \left(H_p^{(\text{sim})}(R)\right)^2}} \tag{13}$$

where $H_p^{(\text{exp})}$ and $H_p^{(\text{sim})}(R)$ denote the heights (i.e., the z-coordinates) of the given and generated image pixels (transformed by R), respectively, and the summation runs over all pixels p.

3.4 Molecular dynamics simulation

To verify the AFM-Fold framework, we performed MD simulations of AK and FlhA_C. For AK, we performed an all-atom MD simulation to compare the distribution of conformational states sampled by MD with those generated by an AlphaFold 3 clone (Protenix [45]) for training the g-CNN. Simulations were conducted using the Amber ff14SB force field [50] with the TIP3P-FB water model and ions [51], starting from the open crystal structure (PDB ID: 4AKE). The simulations were carried out at 300 K and 1 atm using OpenMM [52], and the trajectory was extended to 450 ns to ensure sufficient sampling of the conformational space.

For FlhA_C, due to computational time constraints, we did not conduct all-atom MD simulations. Instead, we generated conformational ensembles using a set of MD surrogate models: AlphaFlow [36] and BioEmu [41], as well as AlphaFold 2 with MSA subsampling [30] using ColabFold [21]. These generative approaches enabled us to obtain a diverse ensemble of conformations, facilitating a comprehensive comparison with the distribution generated by our AFM-Fold framework.

4 Results

4.1 Verification using pseudo-AFM images: adenylate kinase

adenylate kinase (AK) is a well-studied monomeric enzyme known for its large-scale conformational transitions. It is composed of three relatively rigid domains: the central CORE domain (residues 1–29, 68–117, and 161–214), the AMP-binding domain (AMPbd; residues 30–67), and the lid-like ATP-binding domain (LID; residues 118–167). Experimental and computational studies have suggested that upon ligand binding, the enzyme undergoes a transition from an inactive open conformation to an active closed conformation (see Fig. 1 (c)). Here, we evaluated how accurately AFM-Fold reconstructs 3D conformations of AK from artificially created pseudo "experimental" AFM images.

Quality of the conformational ensemble generated by AlphaFold 3 First, we checked the quality of the conformational ensemble generated by AlphaFold 3 used for the training of the g-CNN. Here, the conformations were generated by navigating AlphaFold 3 with CV values on a grid as described in Section 3.3.1. As the CVs, we used the inter-domain distances of all possible combinations of domain pairs (LID–CORE, CORE–AMPbd, and AMPbd–LID). To obtain physically plausible structures as accurately as possible via navigating to the target CVs, we systematically evaluated generated structures using various navigating schedules with different strengths and onset timings, and selected an optimal schedule strategy (see SI section S1 for details). Fig. 2 (b) shows the generated conformations projected to the LID–CORE, and CORE–AMPbd distances (colored by the AMPbd–LID distance), in comparison with Fig. 2 (a), which shows the projection of the conformations sampled by a 450 ns MD simulation. Compared with the MD simulation, the conformations generated by AlphaFold 3 adequately cover a broad conformational space of AK, including both closed (PDB ID: 1AKE) and open (PDB ID: 4AKE) crystal structures. This means that the g-CNN trained on this ensemble can learn the broad relationship between AFM images and the inter-domain distances of AK.

Accuracy of collective variables estimated by the g-CNN Next, we investigated whether the trained g-CNN accurately estimates CV values from given pseudo-AFM images. Here, we estimated CV values from pseudo-AFM images emulated using randomly chosen conformations from the MD simulation trajectory. The pseudo-AFM images were generated with the settings described in Table 2. Fig. 2 (c) shows the distribution of the root squared error in the CV space between the estimated values with the trained g-CNN and the ground-truth values. With the exceptions of some outlier errors greater than 0.500 nm, the trained g-CNN achieves the mean root squared error of 0.254 nm, which is enough resolution to capture intermediate conformational states of AK.

Accuracy of conformations estimated by AFM-Fold Then, we investigated the accuracy of the estimated conformations of our AFM-Fold framework. As a demonstration, we first estimated the closed and open conformations of AK from the pseudo-AFM images emulated from the closed and open crystal structures (the images are shown in Fig. 1 (b)), respectively. Fig. 1 (c) shows the estimated structures by the protocol of AFM-Fold (i.e., g-CNN and AlphaFold 3 navigation). The RMSD between the ground-truth and reconstructed conformations was 0.216 nm for the closed form and 0.176 nm for the open form, respectively. Considering that the RMSD between the closed and open crystal structures is 0.715 nm, AFM-Fold estimation is accurate enough to identify the two states. The c.c. between the AFM images created from the estimated conformations and the given "experimental" pseudo-AFM image is 0.997 for the closed form and 0.998 for the open form, respectively (here, Eq. (13) was used to compute the c.c.).

We then evaluated the RMSDs of the conformations estimated from the pseudo-AFM images emulated from randomly chosen conformations from the MD simulation trajectory. Fig. 2 (e) shows the distribution of the RMSDs between the ground-truth and estimated conformations. Overall, the RMSD values are low and, aside from a small number of outliers, most RMSDs are distributed around the mean value of approximately 0.281 nm. Fig. 2 (d) shows the scatter plot of the RMSD and the root squared errors of the predicted CVs between the ground-truth and estimated conformations. Although a reasonable correlation is observed between the two measures, some scatter was observed in RMSD at similar error levels; that is, even with comparable CV errors, the RMSDs can vary. Upon close inspection of the generated structures, we found that this arises because a given set of current CVs (inter-domain distances) does not always uniquely determine the domain arrangement. Specifically, either hinge-bending (open–close transition) or shear-like (twisted) domain motions can yield similar inter-domain distances, despite resulting in structurally distinct conformations. This degeneracy in the CV–structure mapping explains the occurrence of RMSD outliers even when the predicted CV values are accurate.

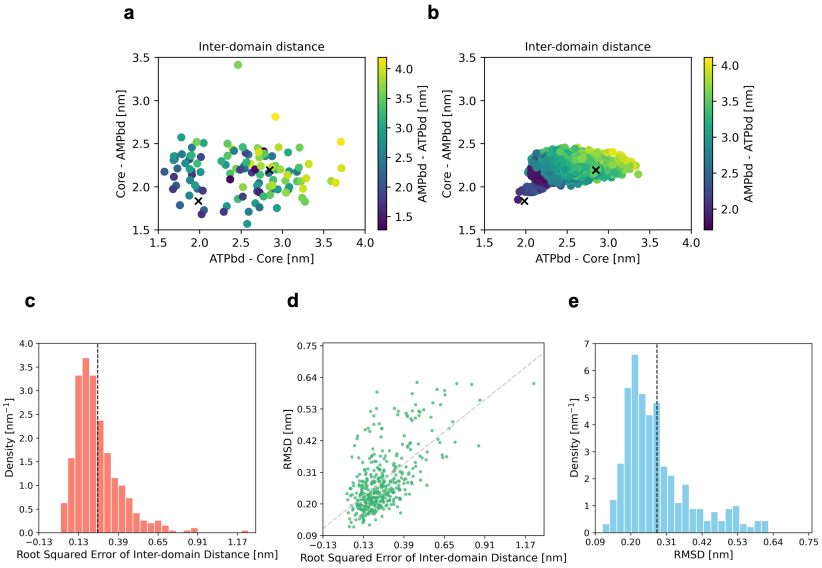

Figure 2: **Training data and estimation accuracy of AFM-Fold**. **(a)** Projection of training data into the inter-domain distance space. The two cross marks represent the open (4AKE) and closed (1AKE) crystal structures from PDB, respectively. **(b)** Projection of a 450 ns MD simulation trajectory into the inter-domain distance space. **(c)** Distribution of the root squared error in the inter-domain distance space between the g-CNN's estimations and the ground-truth. The inter-domain distance was estimated by the trained g-CNN from pseudo-AFM images, which were generated from randomly selected molecular dynamics simulation conformations. The average of the root squared errors is indicated by the broken line (0.254 nm). **(d)** Scatter plot of the root squared error in the inter-domain distance space and the heavy-atom RMSDs from the ground-truth conformation. The heavy-atom RMSD was computed between the conformation generated with AlphaFold 3 navigating using the predicted CV values, and the ground-truth conformation. **(e)** Density of the heavy-atom RMSDs. The average of the RMSD is indicated by the broken line (0.281 nm).

Correlation coefficient and robustness against noise To investigate the applicability of our approach to real experimental data, we next focused on two aspects: (i) the development of evaluation metrics that can be computed without ground-truth structures and (ii) the assessment of robustness in structure estimation accuracy against noise. As for the first aspect, following the previous study [14], we propose to use the c.c. between the given experimental image and a pseudo-AFM image created by rigid-body fitting of the estimated structure to the experimental image, maximizing the c.c. values. Fig. 3 (a) shows the density of the c.c. between the "experimental" images (pseudo-AFM images created from randomly selected structures from the MD data) and pseudo-AFM images created by rigid-body fitting of the estimated structures. With the exceptions of some outlier values, the distribution is centered around the average value of 0.998, indicating that the estimated structures are in good agreement with the given images. Fig. 3 (b) shows the scatter plot of the c.c. and the RMSD. A clear negative correlation is observed between the c.c. and the RMSD, suggesting that, even when ground-truth structures are unavailable and RMSD cannot be computed, as is often the case with real experimental data, the comparison of c.c. values provides a useful indication of the reliability of the estimated structures.

Next, we focused on the second aspect, the assessment of robustness in structure estimation accuracy against noise. So far, we have used noise-free pseudo-AFM images for both the training and inference stages. To assess the robustness of the estimation against noise, we first added noise to "experimental" pseudo-AFM data in the inference stage using Gaussian noise with a standard deviation of 0.3 nm (which is a typical noise level for HS-AFM images [14]). Fig. 3 (c) shows the scatter plot of the c.c. and the RMSD using the noisy "experimental" pseudo-AFM images. Still, a clear negative correlation is observed between the c.c. and the RMSD, suggesting that the c.c. values would work even when noisy real experimental data are used.

Furthermore, for systematic evaluation of the robustness against various noise levels, we system-

atically varied the noise levels added to both the training and the "experimental" pseudo-AFM data. Fig. 3 (d) illustrates examples of pseudo-AFM images at different noise levels, and Fig. 3 (e) shows a heat map of the mean RMSD between the estimated and ground-truth structures over the range of noise levels. Notably, we found that adding noise to the "experimental" pseudo-AFM images consistently led to substantial reductions in prediction accuracy. In contrast, adding noise to the training data alone had a relatively minor effect on the estimation accuracy; indeed, the best performance was achieved when noise-free training data were used. This indicates that, while robustness to noise in the inputs ("experimental" images) is limited, the presence or absence of noise in the training data does not substantially affect accuracy, with noise-free training data providing the most reliable results. Fig. 3 (f) shows a heat map of the mean c.c. between the rigid-body fitted pseudo-AFM images and the "experimental" pseudo-AFM images under the same noise levels as in Fig. 3 (e).

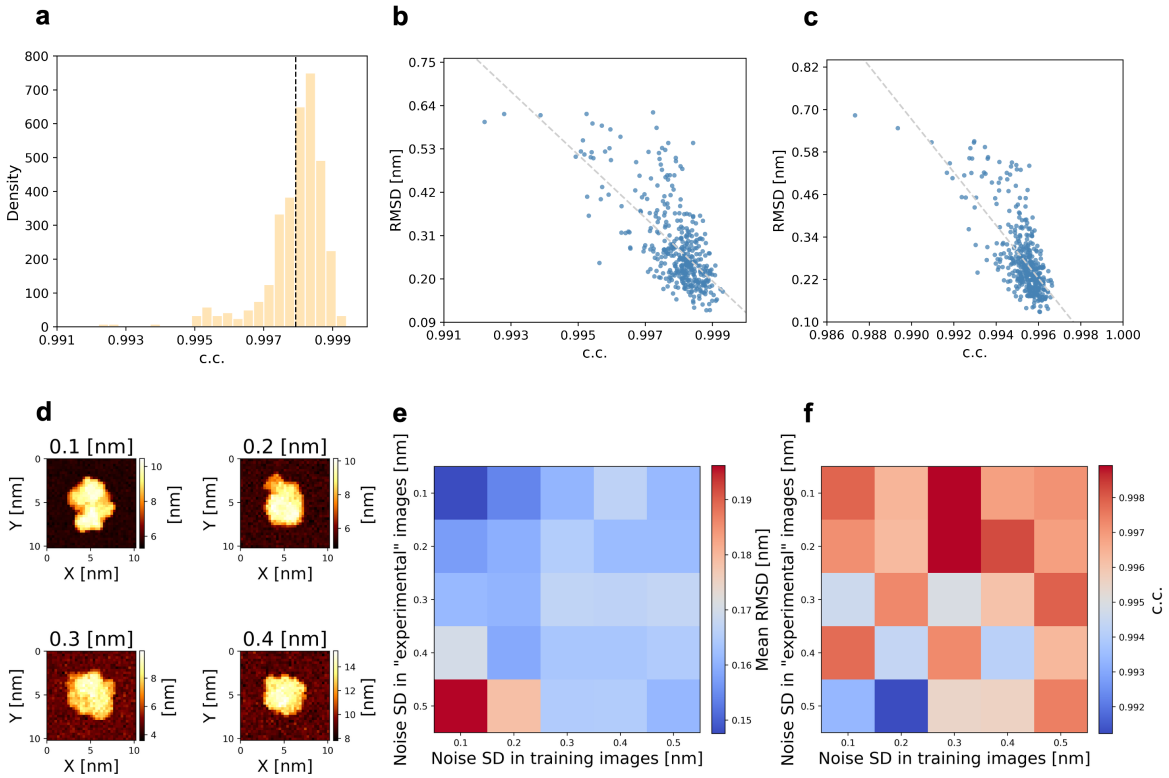

Figure 3: **Correlation coefficients between the given "experimental" images and the pseudo-AFM images generated from the estimated structures, and robustness against noise**. Panels **(a)** and **(b)** present the evaluations under noise-free conditions, whereas panels **(c–f)** evaluate the robustness of the estimations against noise. **(a)** Distribution of correlation coefficients (c.c.) between the given images ("experimental" pseudo-AFM images) and pseudo-AFM images, each of which was generated by a rigid-body fitting of an estimated conformation to the reference image, maximizing the c.c. value. The average of c.c. is indicated by the broken line (0.998). **(b)** Scatter plot of the c.c. and the RMSD. **(c)** Scatter plot of the c.c. and the RMSD using "experimental" pseudo-AFM images contaminated by Gaussian noise with a standard deviation of 0.3 nm. **(d)** Examples of "experimental" pseudo-AFM images used for estimating structures at different noise levels in the investigation of noise robustness. The images were generated from randomly selected structures from the molecular dynamics simulation trajectory and then adding Gaussian noise with various standard deviations. **(e)** Heat map of the mean RMSD between the estimated structures and the ground-truth structures for various noise levels in the training and "experimental" pseudo-AFM data. Each mean RMSD was computed for 20 independent structure estimations. **(f)** Heat map of the mean c.c. between the rigid-body fitted pseudo-AFM images and the "experimental" pseudo-AFM images under the same noise levels as in panel (e).

4.2 Application to real experimental AFM images: a flagellar protein $FlhA_C$

FlhA is a key component of the flagellar protein export system, and its C-terminal domain, $FlhA_C$, helps transport proteins by assisting their binding to the exporter. Structurally, it comprises four domains (A_CD_1, A_CD_2, A_CD_3, and A_CD_4) and a linker to the transmembrane domain ($FlhA_{TM}$) (Fig. 5 (a)). Previous studies using X-ray crystallography and mutational analysis indicate that hinge motions among these domains influence transport activity and motility [28].

Here, to evaluate the applicability of AFM-Fold to real AFM data [53], we estimated the conformations of $FlhA_C$ using AFM-Fold from 159 HS-AFM images of the $FlhA_C$ monomer. It should be noted that, whereas conventional flexible fitting methods for capturing structural changes typically require hours to days of computation per AFM image even with a coarse-grained model [14], AFM-Fold allows estimation of conformations from each AFM image in approximately one minute on a single GPU (considering training time, total computation time is less than the conventional flexible fitting methods). This enabled, for the first time, the processing of as many as 159 AFM images in this study.

Training of the g-CNN Conformations for training the g-CNN were generated by navigating AlphaFold 3 with CV values on a grid as described in Section 3.3.1. As the CVs, we used the inter-domain distances of domain pairs, A_CD_1–A_CD_4, A_CD_2–A_CD_3, and A_CD_2–A_CD_4. The generated ensembles are shown with their MolProbity scores in Fig. S5. The g-CNN was trained on the training data to learn the relationship between AFM images and the CVs.

Evaluating estimated conformations by AFM-Fold Fig. 4 shows the estimated structures of three AFM images selected from the total 159 AFM images. We performed rigid-body fitting of the estimated structures to the real AFM image by exhaustively rotating and translating and finding the pose that maximizes the c.c. value, and then their poses and pseudo-AFM images were drawn for comparison with the real AFM images. As a comparison, we performed the same fitting procedure for the crystal structure (PDB ID: 3A5I) of $FlhA_C$ to the three AFM images (also shown in Fig. 4). Although it is difficult to distinguish by visual inspection alone, in all three cases, the c.c. value between the pseudo-AFM image and the real AFM image was consistently higher for the estimated structure than for the crystal structure. It should be noted, however, that the improvement in c.c. is only slight. This is because the overall c.c. value is largely dominated by whether the molecule is correctly positioned at the center of the image, while the influence of finer structural details such as subtle surface features is relatively minor in comparison, as already shown in the previous verification for AK.

Next, we evaluated the estimated conformations by AFM-Fold using the total 159 images. Fig. 5 (b) shows the comparison of the c.c. values between the estimated structures by AFM-Fold and the crystal structure with the real AFM images. Again, in almost all cases, the c.c. value for the estimated structure was higher than that for the crystal structure. Interestingly, some of the c.c. values of AFM-Fold take higher values when the estimated structures have more open conformations in Fig. 5 (b) (measured by the distance between A_CD_2 and A_CD_4, which serves as a useful indicator to distinguish between open and closed conformations). This might suggest that the flexibility of the inter-domain motions of $FlhA_C$ is well captured by AFM-Fold.

To quantify the flexibility of the inter-domain motions captured by AFM-Fold, we computed the distribution density of the inter-domain distance between A_CD_2 and A_CD_4 for all the estimated structures (fig. 5(c)). For comparison, we also generated conformations using AI-based structure generation models, AlphaFlow [36], BioEmu [41], and AlphaFold 2 with MSA subsampling [30]. These methods are expected to approximate long-time behaviors of $FlhA_C$ that are not accessible by brute-force MD simulations. Interestingly, compared with these models, AFM-Fold tends to estimate more open conformational states. There are three possible explanations for why AFM-Fold tends to estimate more open conformational states than the AI-based structure generation models. First, HS-AFM captures molecular dynamics on a timescale of 1 ms per frame, which may reflect longer-timescale motions than those considered by AlphaFlow or BioEmu, thereby enabling the observation of more open structures. Second, interactions between the molecule and the stage during AFM imaging may stabilize conformations with larger surface areas in contact with the stage, thus favoring more open structures. Third, errors in the structural estimation by AFM-Fold could result in an overrepresentation of highly open conformations with large entropy. These factors together may account for the observed tendency toward open states in our estimations.

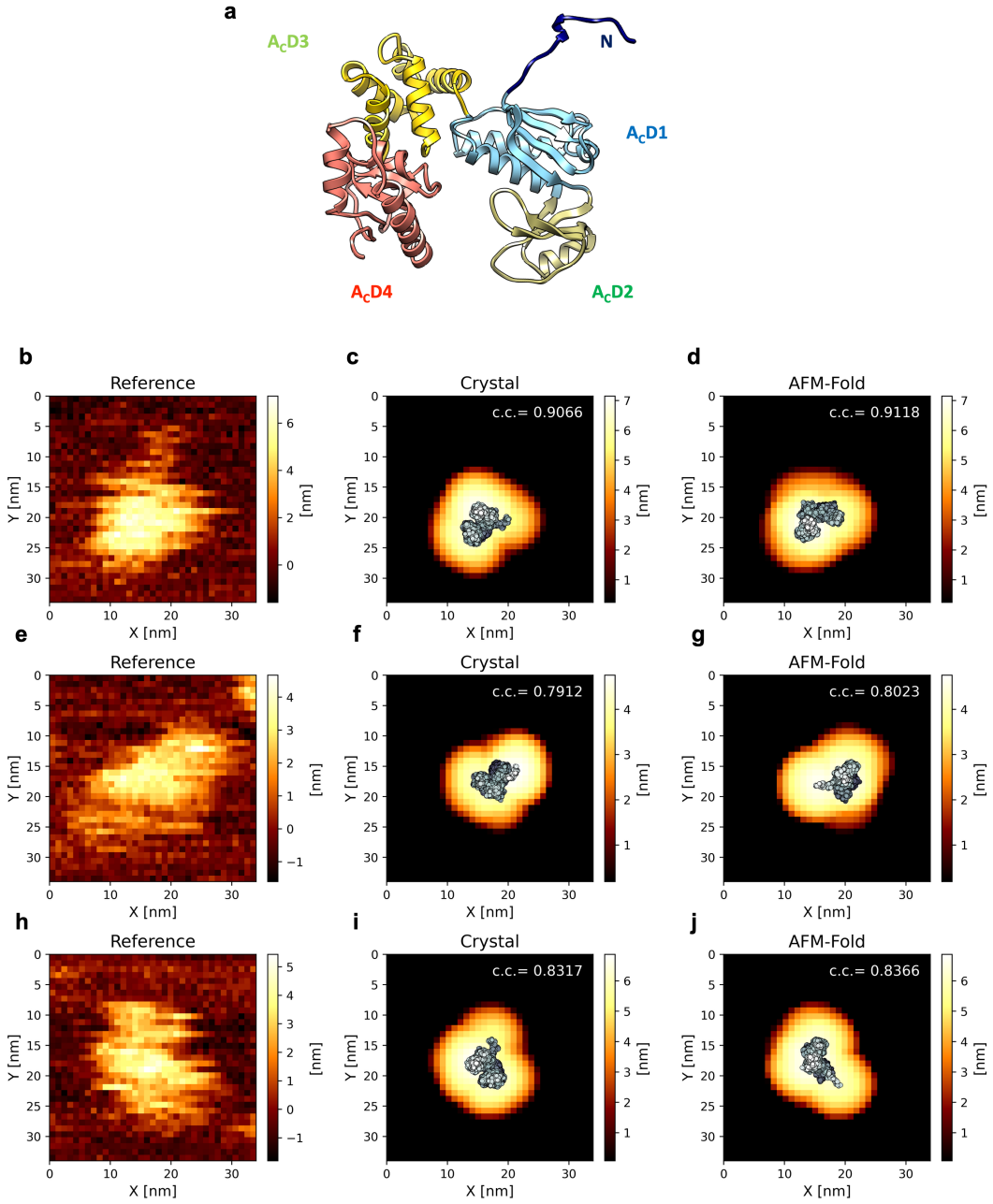

Figure 4: **FlhA$_C$ structure and estimated structures compared with the crystal structure**. **(a)** Crystal structure of FlhA$_C$ (PDB ID: 3A5I). The four domains are colored as follows: dark blue, linker; sky blue, A$_C$D$_1$; green, A$_C$D$_2$; yellow, A$_C$D$_3$; and orange, A$_C$D$_4$. **(b,e,h)** Experimental AFM images selected from real experimental data. **(c,f,i)** Crystal structures fitted to the corresponding experimental AFM images through rigid-body fitting maximizing the c.c. value. Pseudo-AFM images generated from them are shown in the background. **(d,g,j)** Estimated structures by AFM-Fold, fitted to the corresponding experimental AFM images through rigid-body fitting maximizing the c.c. value. Pseudo-AFM images generated from them are shown in the background.

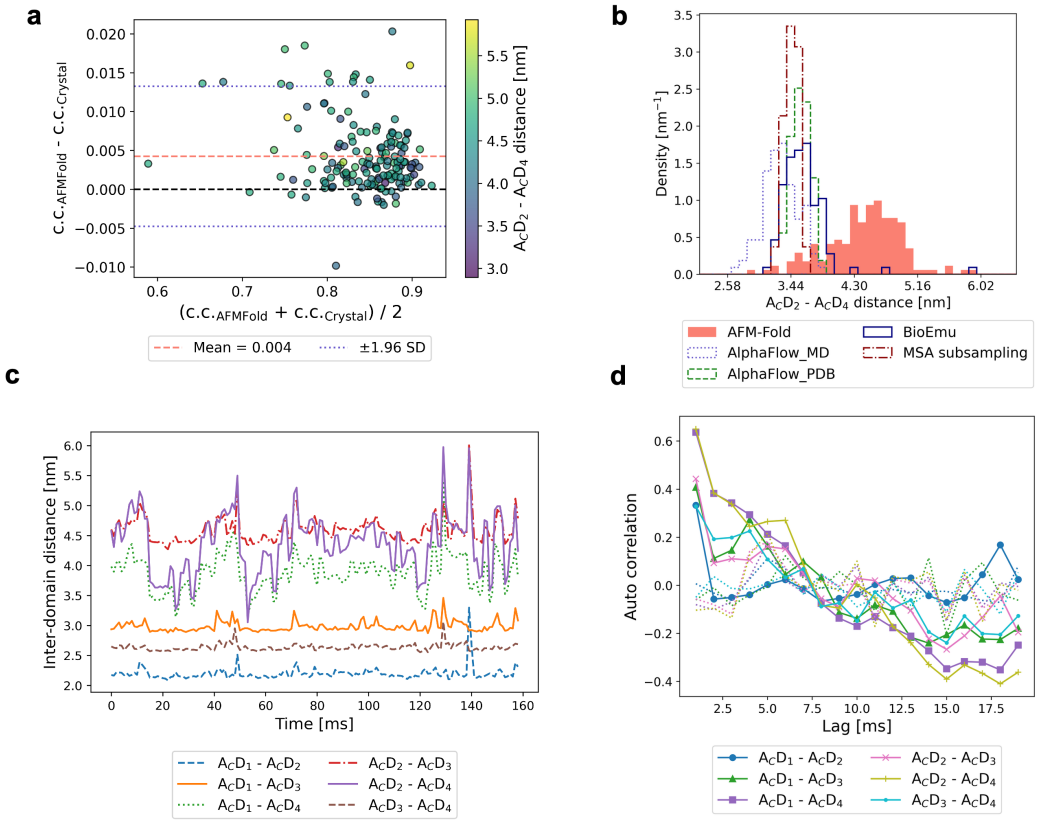

Figure 5: **Analysis of the estimated structures by AFM-Fold**. **(a)** Comparison of correlation coefficients between pseudo-AFM images generated from the AFM-Fold estimations and the experimental AFM images (159 frames) with those of the crystal structure. The x-axis represents the mean of $\mathrm{c.c.}_{\mathrm{AFM-Fold}}$ and $\mathrm{c.c.}_{\mathrm{Crystal}}$, and the y-axis shows their difference ($\mathrm{c.c.}_{\mathrm{AFM-Fold}} - \mathrm{c.c.}_{\mathrm{Crystal}}$). **(b)** Distribution of the inter-domain distances of the AFM-Fold estimations compared with AI-based structure generation models. **(c)** Time series of inter-domain distances estimated by AFM-Fold for the consecutive 159 frames of the experimental AFM images. **(d)** Auto-correlation functions of the inter-domain distance as a function of time lag. The solid line shows the results from AFM-Fold estimations, while the dashed line corresponds to temporally shuffled data.

To verify the third possibility (large estimation errors in AFM-Fold), we investigated the time-series of the estimated structures by AFM-Fold (Fig. 5 (c)). Since the 159 AFM images are a set of consecutive frames as time-series data, we can expect some temporal correlations in the estimated structures if the estimation errors are small. Among the inter-domain distances shown in Fig. 5 (c), the distance between A_CD_2 and A_CD_4 serves as a key indicator distinguishing between open and closed conformations. Examination of the time-series reveals that there are periods where this distance is large and others where it is small, Importantly, this temporal pattern suggests the presence of time correlation in the estimated structures. Despite the fact that AFM-Fold estimates each frame independently without considering temporal information, the observed time correlation in estimated inter-domain distances implies that the estimation errors are smaller than the scale of the observed fluctuations.

The temporal correlation of the inter-domain distances was quantified by the auto-correlation function (Fig. 5 (d)). The solid lines show the auto-correlations from AFM-Fold estimations, while the dashed lines show the auto-correlations from temporally shuffled surrogate data (i.e., the frame order of the inter-domain distances time-series is shuffled). In particular, the auto-correlation for the temporally shuffled data remains low (within ± 0.2) across all lag times, indicating little to no time correlation as expected for randomly ordered data. In contrast, the auto-correlation for the AFM-Fold estimations shows a pronounced positive correlation for lag times up to 5.0 ms and exhibits larger negative values for lag times beyond 15 ms, resulting in oscillatory behavior. Overall, these results demonstrate that AFM-Fold is effective for capturing the molecular conformational dynamics from the time-series of HS-AFM images.

5 Discussion and Conclusions

In this work, we have proposed a computational protocol, AFM-Fold, to predict the 3D conformation of biomolecules from AFM images. AFM-Fold enables rapid and accurate conformation estimation compared to previous approaches. AFM-Fold extracts features from AFM images with a group-invariant (symmetry-aware) CNN, reducing computational cost in two ways. First, it reduces the amount of required training data, because symmetry-aware representations learn efficiently from limited images [25]. Second, it shortens inference time: guiding AlphaFold 3 requires computing $p_t(\phi_{\text{object}}|X_t, a)$; replacing this with $\mathcal{L}_{\text{MSE}}(X_t)$ keeps the computational cost per structure low. In the estimations of this study, per-frame estimation finished within one minute on a single GPU. It has the potential to serve as a powerful tool for interpreting HS-AFM movies comprised of hundreds of frames in both structural and dynamic aspects.

A limitation of AFM-Fold is that it requires a priori definition of CVs as features to be estimated from the AFM image. If it is challenging to define appropriate CVs a priori, one could first perform structural sampling using methods such as AlphaFlow [36], BioEmu [41], or AlphaFold 2 with MSA subsampling [30], and then identify meaningful CVs by applying dimensionality reduction techniques like PCA to extract important features. The AFM-Fold framework is general, and as long as the chosen CVs are differentiable with respect to atomic Cartesian coordinates and can be used for guiding AlphaFold 3, a wide variety of CV types can be incorporated. For example, although we focused on inter-domain distances as CVs in this work, other CVs, such as residue-level distance restraints, can be used to enable AFM-guided reconstruction of finer-grained motions without a significant increase in computational cost.

Another limitation of AFM-Fold is that it infers conformations from a single AFM image and currently does not quantify predictive uncertainty due to image noise or molecular orientation. Extending the framework to estimate uncertainty would make the predictions easier to interpret [54]. Moreover, AFM image sequences constitute time-series data in which poses change only slightly between adjacent frames; incorporating temporal dependencies—for example, with TimeSformer [55] or non-local networks [56]—could further improve fidelity.

Finally, in this work we navigated AlphaFold 3's sampling only using AFM images; consequently, neither structural stability nor explicit physical priors are incorporated. In the future, conditioning the prior (such as Boltz-2's custom potentials [29]) of an ensemble generative model should enable integrated modeling that unifies experimental measurements, physical priors, and computational inference.

6 Code Availability Statement

The code of AFM-Fold is publicly available at http://github.com/matsunagalab/afmfold.

7 Data Availability Statement

Notebooks, scripts, and datasets to reproduce the results of this paper are publicly available at Zenodo [57]. HS-AFM data of $FlhA_C$ are available upon reasonable request to the corresponding author and with the permission of the original authors [53].

8 Acknowledgements

We appreciate Tohru Minamino (The University of Osaka) and Noriyuki Kodera (Kanazawa University) for kindly providing the experimental AFM image data used in this work. This work was supported by JSPS KAKENHI (Grant number: 23H03412), and partly supported by MEXT as "Program for Promoting Researches on the Supercomputer Fugaku" (Development and application of large-scale simulation-based inferences for biomolecules JPMXP1020230119) and used computational resources of supercomputer Fugaku provided by the RIKEN Center for Computational Science (Project IDs: hp230209, hp240215, hp250233).

References

(1) Ando, T.; Uchihashi, T.; Fukuma, T. *Prog. Surf. Sci.* **2008**, *83*, 337–437, DOI: 10.1016/j.progsurf.2008.09.001, (accessed 05/22/2022).

(2) Ando, T., *High-Speed Atomic Force Microscopy in Biology*; NanoScience and Technology, Vol. 17; Springer: 2022, DOI: 10.1007/978-3-662-64785-1.

(3) Villarrubia, J. *J. Res. Natl. Inst. Stand. Technol.* **1997**, *102*, 425, DOI: 10.6028/jres.102.030, (accessed 04/07/2021).

(4) Matsunaga, Y.; Fuchigami, S.; Ogane, T.; Takada, S. *Sci. Rep.* **2023**, *13*, 129, DOI: 10.1038/s41598-022-27057-2.

(5) Ando, T.; Kodera, N.; Uchihashi, T.; Miyagi, A.; Nakakita, R.; Yamashita, H.; Matada, K. *e-J. Surf. Sci. Nanotechnol.* **2005**, *3*, 384–392, DOI: 10.1380/ejssnt.2005.384, (accessed 06/06/2022).

(6) Scheuring, S.; Busselez, J.; Lévy, D. *J. Biol. Chem.* **2005**, *280*, 1426–1431, DOI: 10.1074/jbc.M411334200, (accessed 05/22/2022).

(7) Scheuring, S.; Boudier, T.; Sturgis, J. N. *J. Struct. Biol.* **2007**, *159*, 268–276, DOI: 10.1016/j.jsb.2007.01.021, (accessed 05/22/2022).

(8) Asakawa, H.; Ikegami, K.; Setou, M.; Watanabe, N.; Tsukada, M.; Fukuma, T. *Biophys. J.* **2011**, *101*, 1270–1276, DOI: 10.1016/j.bpj.2011.07.020, (accessed 05/22/2022).

(9) Trinh, M.-H.; Odorico, M.; Pique, M. E.; Teulon, J.-M.; Roberts, V. A.; Ten Eyck, L. F.; Getzoff, E. D.; Parot, P.; Chen, S.-w. W.; Pellequer, J.-L. *Structure* **2012**, *20*, 113–120, DOI: 10.1016/j.str.2011.10.023, (accessed 05/22/2022).

(10) Chaves, R. C.; Teulon, J.-M.; Odorico, M.; Parot, P.; Chen, S.-w. W.; Pellequer, J.-L. **2013**, *26*, 596–604, DOI: 10.1002/jmr.2310, (accessed 05/22/2022).

(11) Amyot, R.; Flechsig, H. *PLoS Comput. Biol.* **2020**, *16*, ed. by Schneidman-Duhovny, D., e1008444, DOI: 10.1371/journal.pcbi.1008444, (accessed 02/11/2022).

(12) Amyot, R.; Marchesi, A.; Franz, C. M.; Casuso, I.; Flechsig, H. *PLoS Comput. Biol.* **2022**, *18*, ed. by Onufriev, A., e1009970, DOI: 10.1371/journal.pcbi.1009970, (accessed 11/16/2025).

(13) Niina, T.; Matsunaga, Y.; Takada, S. *PLoS Comput. Biol.* **2021**, *17*, Publisher: Public Library of Science, e1009215, DOI: 10.1371/journal.pcbi.1009215, (accessed 09/08/2021).

(14) Niina, T.; Fuchigami, S.; Takada, S. *J. Chem. Theory Comput.* **2020**, *16*, 1349–1358, DOI: 10.1021/acs.jctc.9b00901.

(15) Wu, X.; Miyashita, O.; Tama, F. *J. Phys. Chem. B* **2024**, *128*, 9363–9372, DOI: 10.1021/acs.jpcb.4c04189.

(16) Amyot, R.; Miyashita, O.; Wu, X.; Takeda, K.; Kodera, N.; Konno, H.; Tama, F.; Flechsig, H. *ACS Nano* **2025**, *19*, 34065–34079, DOI: 10.1021/acsnano.5c10073, (accessed 11/16/2025).

(17) Dasgupta, B.; Miyashita, O.; Tama, F. *BBA–Gen. Subjects* **2020**, *1864*, 129420, DOI: 10.1016/j.bbagen.2019.129420, (accessed 07/04/2024).

(18) Dasgupta, B.; Miyashita, O.; Uchihashi, T.; Tama, F. *Front. Mol. Biosci.* **2021**, *8*, Publisher: Frontiers, DOI: 10.3389/fmolb.2021.704274, (accessed 07/04/2024).

(19) Jumper, J.; Evans, R.; Pritzel, A., et al. *Nature* **2021**, *596*, 583–589, DOI: 10.1038/s41586-021-03819-2.

(20) Ahdritz, G.; Bouatta, N.; Floristean, C.; Kadyan, S.; Xia, Q.; Gerecke, W.; O'Donnell, T. J.; Berenberg, D.; Fisk, I.; Zanichelli, N., et al. *Nat. Methods* **2024**, *21*, 1514–1524, DOI: 10.1038/s41592-024-02272-z.

(21) Mirdita, M.; Schütze, K.; Moriwaki, Y.; Heo, L.; Ovchinnikov, S.; Steinegger, M. *Nat. Methods* **2022**, *19*, 679–682, DOI: 10.1038/s41592-022-01488-1.

(22) Wohlwend, J.; Corso, G.; Passaro, S.; Getz, N.; Reveiz, M.; Leidal, K.; Swiderski, W.; Atkinson, L.; Portnoi, T.; Chinn, I.; Silterra, J.; Jaakkola, T.; Barzilay, R. *bioRxiv* **2024**, DOI: 10.1101/2024.11.19.624167.

(23) Maddipatla, A.; Sellam, N. B.; Bojan, M.; Vedula, S.; Schanda, P.; Marx, A.; Bronstein, A. M. *arXiv* **2025**, Preprint, https://arxiv.org/abs/2502.09372.

(24) Raghu, R. et al. *arXiv* **2025**, Preprint, https://arxiv.org/abs/2506.04490.

(25) Weiler, M.; Cesa, G. **2019**, https://proceedings.neurips.cc/paper/2019/hash/45d6637b718d0f24a237069fe41 Abstract.html.

(26) Müller, C. W.; Schulz, G. E. *J. Mol. Biol.* **1992**, *224*, 159–177, DOI: 10.1016/0022-2836(92)90582-5.

(27) Müller, C. W.; Schlauderer, G. J.; Reinstein, J.; Schulz, G. E. *Structure* **1996**, *4*, 147–156, DOI: 10.1016/S0969-2126(96)00018-4.

(28) Saijo-Hamano, Y.; Imada, K.; Minamino, T.; Kihara, M.; Shimada, M.; Kitao, A.; Namba, K. *Mol. Microbiol.* **2010**, *76*, 260–268, DOI: 10.1111/j.1365-2958.2010.07097.x.

(29) Passaro, S.; Corso, G.; Wohlwend, J.; Reveiz, M.; Thaler, S.; Somnath, V. R.; Getz, N.; Portnoi, T.; Roy, J.; Stark, H.; Kwabi-Addo, D.; Beaini, D.; Jaakkola, T.; Barzilay, R. *bioRxiv* **2025**, DOI: 10.1101/2025.06.14.659707.

(30) Del Alamo, D.; Sala, D.; Mchaourab, H. S.; Meiler, J. *eLife* **2022**, *11*, e75751, DOI: 10.7554/eLife.75751.

(31) Wayment-Steele, H. K.; Ojoawo, A.; Otten, R.; Apitz, J. M.; Pitsawong, W.; Hömberger, M.; Ovchinnikov, S.; Colwell, L.; Kern, D. *Nature* **2023**, *625*, 832–839, DOI: 10.1038/s41586-023-06832-9.

(32) Monteiro da Silva, G.; Cui, J. Y.; Dalgarno, D. C.; Lisi, G. P.; Rubenstein, B. M. *Nat. Commun.* **2024**, *15*, 2464, DOI: 10.1038/s41467-024-46715-9.

(33) Kalakoti, Y.; Wallner, B. *Commun. Biol.* **2025**, *8*, 373, DOI: 10.1038/s42003-025-07791-9.

(34) Noé, F.; Olsson, S.; Köhler, J.; Wu, H. *Science* **2019**, *365*, eaaw1147, DOI: 10.1126/science.aaw1147.

(35) Jing, B.; Erives, E.; Pao-Huang, P.; Corso, G.; Berger, B.; Jaakkola, T. *arXiv* **2023**, https://arxiv.org/abs/2304.02198.

(36) Jing, B.; Berger, B.; Jaakkola, T. **2024**, *896*, 22277–22303, https://arxiv.org/abs/2402.04845.

(37) Zheng, S. et al. *Nat. Mach. Intell.* **2024**, *6*, 558–567, DOI: 10.1038/s42256-024-00837-3.

(38) Cheng, K.; Liu, C.; Su, Q.; Wang, J.; Zhang, L.; Tang, Y.; Yao, Y.; Zhu, S.; Qi, Y. *arXiv* **2024**, https://arxiv.org/abs/2408.12419.

(39) Jing, B.; Stärk, H.; Jaakkola, T.; Berger, B. **2024**, *37*, 40534–40564, https://proceedings.neurips.cc/paper_files/paper/2024/file/478b06f60662d3cdc1d4f15d4587173a-Paper-Conference.pdf.

(40) Jin, Y.; Huang, Q.; Song, Z.; Zheng, M.; Teng, D.; Shi, Q. *J. Chem. Theory Comput.* **2025**, *21*, 3288–3296, DOI: 10.1021/acs.jctc.4c01620.

(41) Lewis, S. et al. *Science* **2025**, eadv9817, DOI: 10.1126/science.adv9817.

(42) Liu, Y.; Sahoo, A.; Yu, Z.; Lin, G.; Chen, M.; Hanson, S. M. *bioRxiv* **2024**, DOI: 10.1101/2024.10.04.616517.

(43) Liu, Y.; Yu, Z.; Lindsay, R. J.; Lin, G.; Chen, M.; Sahoo, A.; Hanson, S. M. *PRX Life* **2025**, *3*, 023013, DOI: 10.1103/PRXLife.3.023013.

(44) Levy, A.; Chan, E. R.; Fridovich-Keil, S.; Poitevin, F. P.; Zhong, E. D.; Wetzstein, G. **2025**, https://openreview.net/forum?id=UYZRaUCLAg.

(45) Team, B. A. A. et al. *bioRxiv* **2025**, DOI: 10.1101/2025.01.08.631967.

(46) Cohen, T. S.; Welling, M. **2016**, *48*, 2990–2999, https://proceedings.mlr.press/v48/cohenc16.html.

(47) Cohen, T. S.; Welling, M. **2017**, https://openreview.net/forum?id=rJQKYt5ll.

(48) Weiler, M.; Geiger, M.; Welling, M.; Boomsma, W.; Cohen, T. S. **2018**, *31*, https://papers.nips.cc/paper_files/paper/2018/hash/488e4104520c6aab692863cc1dba45af-Abstract.html.

(49) Karras, T.; Aittala, M.; Aila, T.; Laine, S. **2022**, 26565–26577, https://proceedings.neurips.cc/paper_files/paper/2022/file/a98846e9d9cc01cfb87eb694d946ce6b-Abstract.html.

(50) Maier, J. A.; Martinez, C.; Kasavajhala, K.; Wickstrom, L.; Hauser, K. E.; Simmerling, C. **2015**, *11*, Publisher: American Chemical Society, 3696–3713, DOI: 10.1021/acs.jctc.5b00255, (accessed 11/16/2025).

(51) Wang, L.-P.; Martinez, T. J.; Pande, V. S. *J. Phys. Chem. Lett.* **2014**, *5*, Publisher: American Chemical Society, 1885–1891, DOI: 10.1021/jz500737m, (accessed 11/16/2025).

(52) Eastman, P. et al. *J. Phys. Chem. B* **2024**, *128*, Publisher: American Chemical Society, 109–116, DOI: 10.1021/acs.jpcb.3c06662, (accessed 01/24/2024).

(53) Inoue, Y.; Ogawa, Y.; Kinoshita, M.; Terahara, N.; Shimada, M.; Kodera, N.; Ando, T.; Namba, K.; Kitao, A.; Imada, K.; Minamino, T. *Structure* **2019**, *27*, 965–976.e6, DOI: 10.1016/j.str.2019.03.017, (accessed 10/15/2025).

(54) Kendall, A.; Gal, Y. **2017**, *30*, https://proceedings.neurips.cc/paper/2017/hash/2650d6089a6d640c5e85b2b88265dc2b-Abstract.html.

(55) Bertasius, G.; Wang, H.; Torresani, L. *Proc. Mach. Learn. Res.* **2021**, *139*, 813–824, https://proceedings.mlr.press/v139/bertasius21a.html.

(56) Wang, X.; Girshick, R.; Gupta, A.; He, K. **2018**, 7794–7803, DOI: 10.1109/CVPR.2018.00813.

(57) Kawai, T.; Matsunaga, Y., DOI: 10.5281/zenodo.17597490, (accessed 11/13/2025).