Enhancing AFM Image Analysis and Prediction through Machine Learning and Style Translation

Midterm Report

Jie Huang
Supervisor: **Adam S. Foster**

October 2024

Aalto University
School of Science
Department of Applied Physics
Surfaces and Interfaces at the Nanoscale (SIN)

Contents

1 Introduction

1.1 Atomic force microscopy

Atomic Force Microscopy (AFM), invented in 1986 [1], is a powerful tool used to investigate the structure of atomic-scale structures. Compared to other techniques such as nuclear magnetic resonance (NMR), and mass spectrometry, which require reconstruction from indirect information, AFM offers a direct method for acquiring high-resolution topographical images of material surfaces. AFM has several operational modes, including contact mode, tapping mode, and non-contact mode, each optimized for specific applications. Among the various AFM methods, non-contact AFM (NC-AFM) has emerged as a crucial tool for characterizing nanostructures on the atomic scale, and the first NC-AFM image was produced in 1987 [2]. Since its tip never touches the sample, NC-AFM doesn't disturb or destroy the sample.

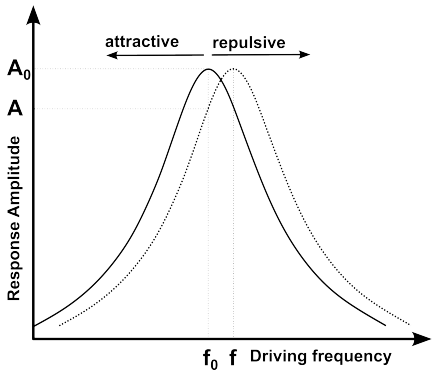

Figure 1: Resonance curve for a harmonic oscillator (solid line) and under the influence of external force (dashed line) [3].

NC-AFM can be achieved by using the frequency-modulation method to measure the frequency shift of the AFM tip [4]. The tip is attached to a cantilever, acting as the oscillator, whose oscillation frequency or driving frequency f is controlled by a feedback circuit. Driving frequency is independent of the cantilever's natural frequency f_0. However, as the driving frequency gets closer and closer to the natural frequency f_0, the response amplitude of the oscillation rises dramatically and reaches a maximum when the driving frequency matches the cantilever's natural frequency f_0. The solid line in Figure 1 shows the relation between the response amplitude of the cantilever and the driving frequency when there is no external force applied to the tip. However, as the tip approaches the surface, the external force will shift the resonance curve as shown in the dashed line in Figure 1. Initially, the driving frequency is set at the natural frequency f_0 with amplitude A_0. When the external force shifts the resonance, the amplitude decreases from A_0 to A. The feedback circuit is designed to maintain the oscillation amplitude, so it would adjust the driving frequency from f_0 to a new frequency f. Then the frequency shift $\Delta f = f - f_0$ is recorded for NC-AFM

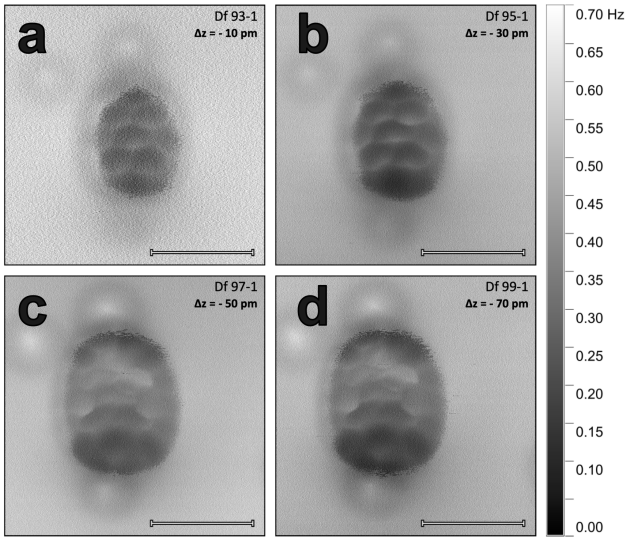

Figure 2: CO tip NC-AFM images of PTCDA on Calcite(104) at different height at T = 5 K(Image source: Jonas Heggemann, Paul Laubrock, Tim Dierker and Philipp Rahe, 2024.)

imaging. Besides, functionalized tips, such as carbon monoxide (CO) and xenon (Xe), can be employed to further enhance NC-AFM imaging capabilities [5].

Figure 2 shows NC-AFM images of PTCDA on Calcite (104) at various tip heights. Since the force that the tip experiences changes with its height, AFM images at different heights are distinct from each other. As the tip gradually approaches the sample, as shown in Figure 2a to d, the central part becomes brighter, indicating an increase in the frequency shift. We can see five rings in Figure 2a,b when the tip is close. These features make us believe these rings correspond to the five benzene rings in the PTCAD molecule. However, other rings (functional groups) connected to the oxygen atoms are only revealed in lower-height images as shown in the lower and upper parts in Figure 2c,d. Why are the features of the functional group not included in the higher-height AFM images where benzene rings of the same molecule can be clearly revealed? We encounter difficulties when trying to answer questions like why an AFM image looks like this. Therefore, these uncertainties drive us to use advanced simulation tools to explore how samples are adsorbed onto surfaces and to help us explain the AFM images or even further guide AFM experiments.

1.2 Simulations and the Probe Particle Model

In addition to AFM experiments, we can also simulate the systems of samples and surfaces, and then generate simulation AFM images. AFM simulations play a crucial role in elucidating the mechanisms behind the high-resolution capabilities of functionalized AFM. They also provide powerful tools for studying nano surfaces from another perspective, offering low-cost and highly efficient approaches. To generate AFM images at various heights, we employ

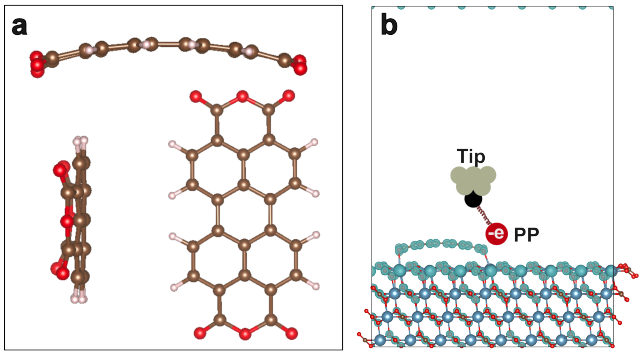

Figure 3: (a) Relaxed sample structure calculated by Density Functional Theory. (b) Probe Particle AFM Model.

a highly efficient graphical processing implementation of the Probe Particle Model (PPM) [6, 7, 8], which allows us to rapidly obtain 2D simulated AFM (PPAFM) stacks.

To simulate the PPAFM images, we first simulate the system of the sample and surface using Density Functional Theory (DFT) [9, 10, 11]. As an example, a relaxed PTCAD molecule on Calcite (104) is shown in Figure 3a in three perspectives, where the Calcite surface is not plotted for clean visualization. From the results from DFT simulations, it's obvious to tell that that the PTCDA molecule is no longer a planar structure. Upon adsorption onto the Calcite surface, the PTCDA has become bent or distorted, which explains why the features of the functional group don't appear when benzene rings are clearly revealed as the tip gradually closes to the PTCDA sample.

Next, we use the relaxed system of sample and surface as well as the corresponding electrostatic potential calculated from DFT as the input of PPM to generate PPAFM images. As shown in Figure 3b, PPM simulates a CO molecule on the tip, so the tip is negative charged with the typical values in the range -0.05e to -0.10e. Given that the scanning speed is much slower than the timescales of molecular relaxation processes, we assume the probe particle is fully relaxed at every point of the tip oscillation trajectory. Therefore, the total force $\mathbf{F}_{\mathrm{PP}}(\mathbf{r})$ on the PP at any given position $\mathbf{r}$ is considered to be zero. Additionally, we assume the tip does not interact with the sample and surface. As a result, the force $\mathbf{F}_{\mathrm{T}}(\mathbf{r})$ experienced by the tip is equal in magnitude but opposite in direction to the force it exerts on the PP. Then the vertical component of the force $\mathbf{F}_{\mathrm{T}}(\mathbf{r})$ is converted to the frequency shift for NC-AFM imaging.

When calculating the force of PP exerted by the sample and surface, two types of forces are considered: van der Waals force and electrostatic force. In practice, we obtain the corresponding energy surfaces in 3D space, add them together, and then calculate the negative change in the total energy with respect to the change in position $\mathbf{r}_i$ of the particle to obtain the force given by sample and surface, which is defined as $\mathbf{F}_i = -\nabla_{\mathbf{r}_i} E$.

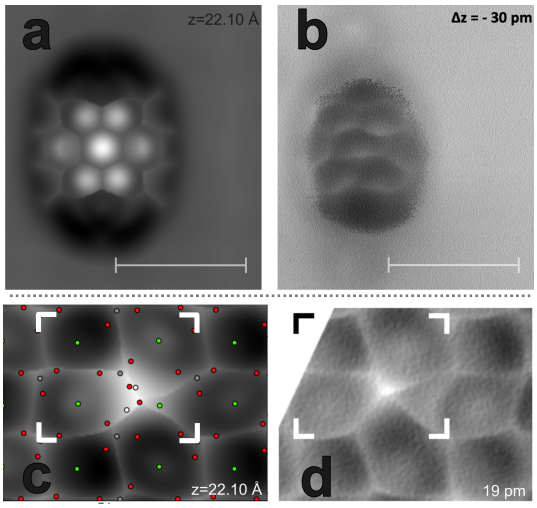

Figure 4: The comparison of simulated PPAFM and experimental AFM images for the systems of PTCDA (a, b) and water molecule (c, d) on calcite surface.

Once we know the force on the PP given by the sample and surface, we know the force on the tip $\mathbf{F}_{\mathrm{T}}(\mathbf{r})$. Finally, we can turn $\mathbf{F}_{\mathrm{T}}(\mathbf{r})$ into frequency shift value. Moving the tip position in 3D space, we can obtain the whole stack of 2D NC-AFM images.

Figure 4a shows a PPAFM image corresponding to the sample and surface configurations in Figure 3 at one specific tip height. By comparing to the real AFM image as shown in Figure 4b, we see that the PPAFM matches the real experimental image pretty well, which verifies the structure given by the DFT calculations in Figure 3a. Additional comparisons between PPAFM and experimental AFM of a water molecule on a Calcite surface are presented in Figures 4c and d. The PP model effectively reproduces the characteristics of the experimental images.

In summary, DFT calculations give us a relaxed structure of the sample and surface. Then, the PPM model provides the map from relaxed structure to PPAFM images, through which we can compare with the experimental AFM images. When the images of PPAFM and AFM are close enough, we believe that the structure calculated from DFT is the structure of the experimental AFM images. This demonstrates that DFT and PPM are very useful tools to help us understand how samples are absorbed on surfaces.

1.3 Automated AFM image interpretation

In the previous example, we show an example of determining how a single PTCDA molecule is absorbed on Calcite surface. However, when samples are getting more complex like there are some molecule clusters on the surface, interpreting the AFM images would become extremely challenging. The difficulty could come from the following two folds: (1) The complexity of the sample itself. It is more challenging to interpret AFM images of complex

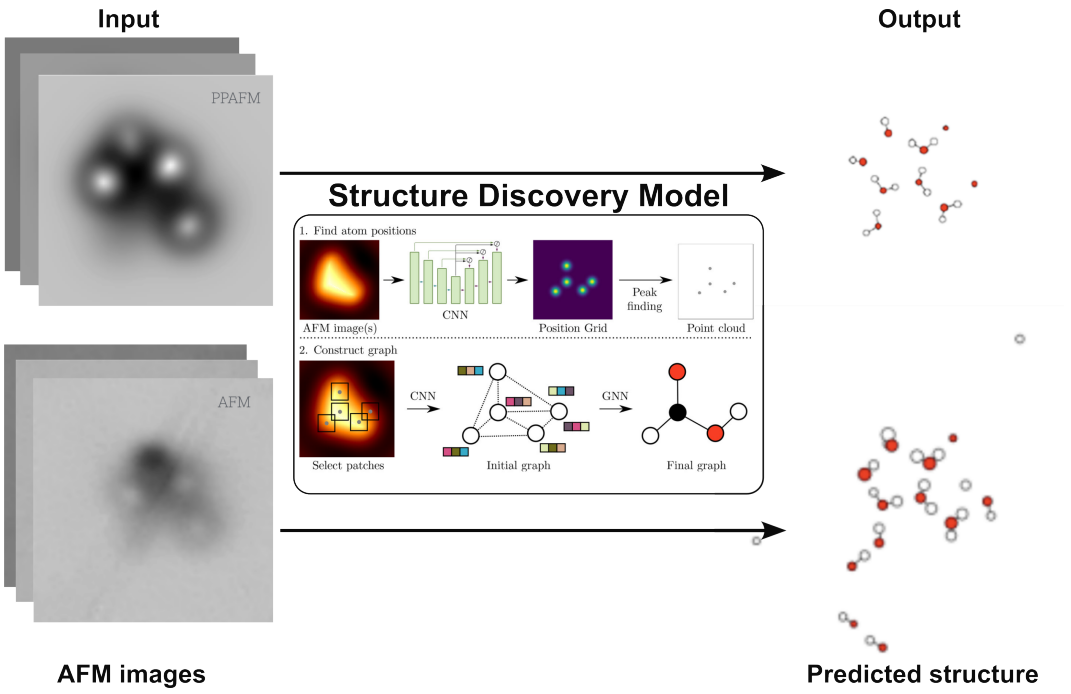

Figure 5: The inputs and outputs of a structure discovery model trained on the PPAFM dataset.

samples; (2) Using DFT to get the initial guess of possible configures can be very expensive, as it typically needs lots of trial and error loops to find the correct configuration.

Is there a better way to help us guess the sample configurations on the surfaces more quickly and directly? The answer is yes. PPM allows us to map a configuration of sample and surface to PPAFM images. With the assistance of machine learning, we can learn a map from PPAFM images to a configuration if we train a model with a large number of pairs of (PPAFM images, configuration). Once a well-trained model is obtained, we can input AFM images into this trained model to ask it to give the corresponding configuration prediction.

Figure 5 shows the current workflow of the structure discovery machine learning model. The structure discovery model is trained on the multi-layers of 2D PPAFM images and their corresponding sample atomic 3D structures. It's worth emphasizing that one training sample contains multi-layers of 2D images and a corresponding 3D atomic sample structure serves as the label. This structure discovery model is trained to learn the mapping from a stack of 2D AFM images to a 3D atomic configuration.[12, 13]. A well-trained model can give very good predictions of a given stack of 2D PPAFM images. As can be seen from Figure 5, the prediction of the model is represented as a graph containing atoms and their bond connections as well as their coordinates within the image and elemental identification.

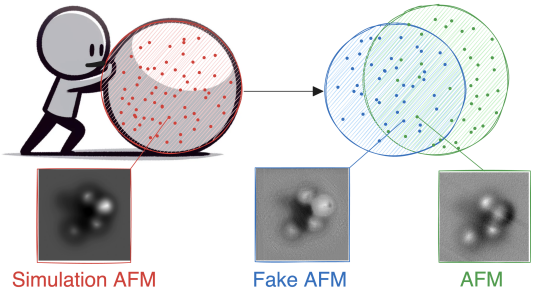

Figure 6: Hypothesis: If we can make some modifications to the original PPAFM images to let them have the features of real experimental images, or translate the simulation style to the real experimental style. Then the newly generated images are combined with the the sample configurations corresponding to the original PPAFM images to construct the training set to train the machine learning models. Since the machine learning models have 'seen' the experimental-style images, the model should logically perform better than the model trained on simulation images.

Even though this model is trained by utilizing the simulation PPAFM images, we can still input the experimental AFM images to the trained model and get the structure prediction. As shown in Figure 5, compared to its prediction of PPAFM input, the model hallucinates some atoms around the main water cluster when inputting the real AFM images, which is unreasonable, indicating that the model trained on simulation PPAFM images cannot handle the real experimental AFM well.

However, it is the ultimate goal to apply these models to real experimental AFM images and to discover their sample structure. As you might have noticed, a very critical problem arises when we use the trained model. Although simulated images resemble real AFM images, they are still distinguished from each other, i.e., the 'style' is different. Simulated PPAFM looks smoother and 'perfect', while experimental images have more noise and artifacts. Therefore, when inputting real AFM images to the model trained on the simulation dataset, the model performance decreases.

1.4 Motivation and Hypothesis

We believe that there is a distribution shift between the training PPAFM images and the experimental AFM images in real applications – when inputting the experimental AFM images, we cannot expect these machine learning models to perform as well as the case when inputting the simulated PPAFM images which the model has 'seen' in training. This problem happens in every machine learning model where the application data are different from the data in training.

As shown in Figure 6, a single PPAFM image can be viewed as a sample from a high dimensional distribution related to the simulation data. However, this high-dimensional

distribution is different from the experimental distribution. Is there a way that can increase the overlap of the two distributions? Can we generate some other datasets that are similar to the real experiment dataset? Or go even further, can we learn the experimental AFM style and apply it to our simulation dataset? These questions are important because if we can get more data that are much more like the real AFM data, we can use this dataset to train a more robust model, then a better model to predict the properties of the AFM sample. We believe that if the answer is yes, we can fill the gap between the simulation and experiment data.

2 Methods and Material

Then the next question would be how to find the right way to modify our original PPAFM images to make them have the features of real AFM images, i.e., to translate PPAFM style to real AFM style. CycleGAN is a machine learning framework to learn two generators to translate image style between two image domains [14]. CycleGAN learns where to make modifications automatically, keeping the irrelevant parts the same, so it inspires us to explore the translations between PPAFM and real AFM. Since we don't need to paired images in two image domains in CycleGAN training, we only need to collect two sets of images in two domains. We can obtain as many PPAFM images as we want and collect a considerable amount of experimental AFM images. Therefore, it's possible to train a CycleGAN model for PPAFM and AFM.

2.1 Style translation between PPAFM and AFM using CycleGAN

Figure 7 illustrates the framework of CycleGAN. There are two generators G and F corresponding to two mapping functions G: $A \rightarrow B$ and F: $B \rightarrow A$, and associated adversarial discriminators D_{A} and D_{B}, which encourages G to translate A into outputs indistinguishable from domain B, and vice versa for D_{A} and F.

For the forward generator G, an image-to-image neural network, its goal is to generate realistic AFM images. To train G, another neural network D, whose input is an image and output is a number, is needed to evaluate the generated images. The goal of G is to change its neural network parameters to minimize $E_{x\sim p_{data}(x)}[\log(1-D_Y(G(x)))$, where x is a PPAFM image. In contrast, D is designed to reward the real AFM images and punish the generated fake AFM images. Hence to train D, we need to tune the parameters of D to maximize $E_{y\sim p_{data}(y)}[\log D_Y(y)]+E_{x\sim p_{data}(x)}[\log(1-D_Y(G(x)))]$. Hence, we can combine the loss function for both G and D as follows:

$$\mathscr{L}_{GAN}(G,D_Y,X,Y)=E_{y\sim p_{data}(y)}[\log D_Y(y)]+E_{x\sim p_{data}(x)}[\log(1-D_Y(G(x)))] \qquad (1)$$

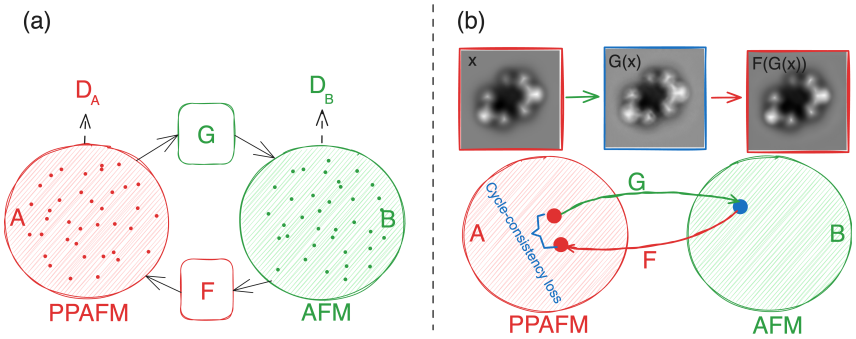

Figure 7: (a) CycleGAN includes two mapping functions G: $A \rightarrow B$ and F: $B \rightarrow A$, and associated adversarial discriminators D_{A} and D_{B}, which encourages G to translate A into outputs indistinguishable from domain B, and vice versa for D_{A} and F. (b) Cycle consistency ensures that converting from one domain to another and back again returns to the original starting point.

Then, the optimal neural network parameters in G and D is as follows:

$$G^*, D_Y^* = \arg\min_G \max_D \mathscr{L}_{GAN}(G, D_Y, X, Y)$$

A similar expression can be determined for the backward translation. Another critical mechanism in CycleGAN is called cycle consistency as shown in Figure 7b. Cycle consistency ensures that converting from one domain to another and back again returns to the original starting point, i.e., $x \rightarrow G(x) \rightarrow F(G(x)) \approx x$. This mechanism is close to language translation: when you translate one sentence to another language, and then translate the results using the reverse translator, you should get a similar sentence to your initial input sentence. So the following term is added to the loss function:

$$\mathscr{L}_{cyc}(G,F) = E_{x\sim p_{data}(x)}[\|F(G(x)) - x\|_1] + E_{y\sim p_{data}(y)}[\|G(F(x)) - y\|_1] \tag{2}$$

Therefore, the total loss of these four neural networks is as follows:

$$\mathscr{L}(G,F,D_X,D_Y) = \mathscr{L}_{GAN}(G,D_Y,X,Y) + \mathscr{L}_{GAN}(F,D_X,Y,X) + \lambda \mathscr{L}_{cyc}(G,F) \tag{3}$$

where λ serves as the loss weight for the cycle consistency mechanism. A large value would lead to more conservative generators, with the final goal of finding the optimal parameters for G and F:

$$G^*, F^* = \arg\min_{G,F} \max_{D_X,D_Y} \mathscr{L}(G,F,D_X,D_Y) \tag{4}$$

The simulation data comes from our previous work [12], which is made using the implementation of PPM and available at the GitHub repository https://github.com/SINGROUP/Graph-AFM. In this study, 609 PPAFM images for the systems of water clusters on gold surfaces and 510 experimental AFM images for different systems are used to train our CycleGAN model.

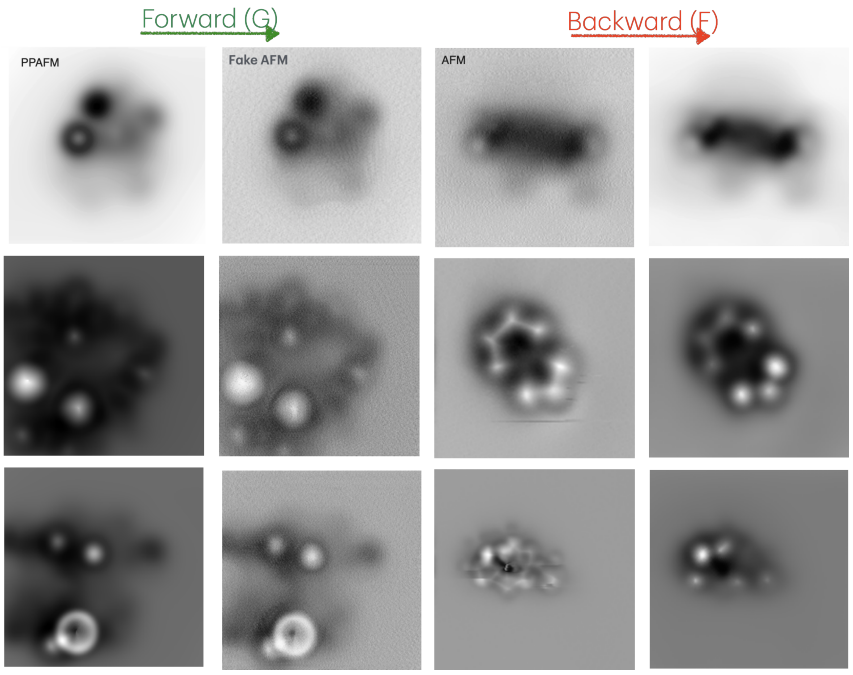

Figure 8: The forward and backward translation. The trained forward generator receives PPAFM images as an input and outputs the corresponding images (fake AFM) which are expected to have the features of real AFM images. The backward generator does the reverse translation to turn real AFM images into PPAFM-style images (fake PPAFM).

3 Results and discussions

3.1 Style translations and its evaluations

As shown in Figure 8, the trained cycleGAN model can learn the features of both PPAFM and real AFM. For the forward translation, the network G can produce images with realistic features like noise. Fake AFM images' background intensity changes and more experimental features are added compared to the original simulation PPAFM images. For the backward generator F, the output images resemble the simulation PPAFM images, where the noise and artifact are removed from the real AFM inputs, which holds the potential for removing the artifacts or denoising experimental images.

Nevertheless, it is relatively hard to tell whether the style translations are successful by eye since we don't have specific rules to define which features belong to simulation images and which image features belong to real AFM.

To quantify how good these style translations are, we first trained a machine expert to help us give high scores to real AFM images and low scores to PPAFM images. As shown in Figure 9a, the machine expert's input is an image, and its output is a number. The output is defined as the authenticity of an AFM image, which indicates the likelihood that this machine 'thinks' a given image is a real AFM image from experiments. We label the real AFM images to 1,

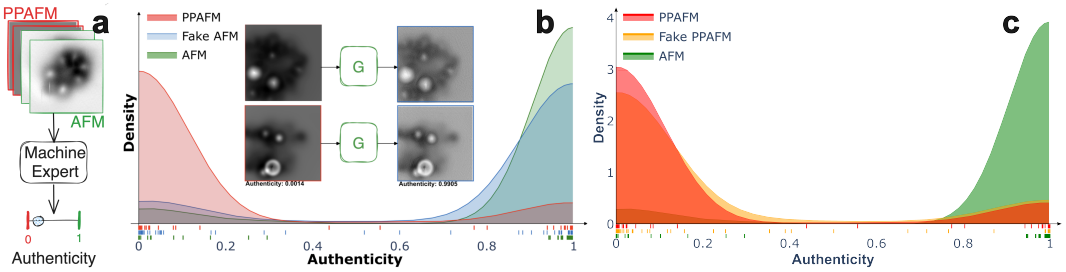

Figure 9: Style translation evaluations from the perspective of a trained expert. (a) The machine expert is a neural network whose input is an image and output is a number. The machine expert is trained to give a value of 1 to real AFM images and 0 to the PPAFM images. (b) Forward style translation evaluation. (c) Backward style translation evaluation.

and PPAFM images to 0, and train the machine expert in a supervised way. Then, this trained machine expert is used to give scores to newly generated fake AFM and fake PPAFM from the output of forward generator G and backward generator F respectively by giving inputs of PPAFM and real AFM images to them.

The authenticity distribution density of PPAFM images and real AFM images are colored red and green respectively in Figure 9. To evaluate the forward translation quality, we first input PPAFM images to the forward translation G and obtain the corresponding outputs, fake AFM images, and then we show the newly generated fake AFM to the machine expert to test their authenticity. The authenticity distribution density for fake AFM images is plotted as blue in Figure 9b. Through forward translation G, the red PPAFM distribution is shifted to the blue fake AFM distribution that is closer to the real AFM distribution when compared to the original PPAFM distribution. In other words, the forward generator indeed 'pushes' the original PPAFM distribution to a new distribution that is closer to the real experimental AFM distribution. Similarly, the backward translation can turn real AFM distribution into a new distribution as shown in the orange plot in 9c. Indeed, we are not 100% successful in style translation, since the new distribution is not completely overlapping the real AFM distribution. However, we still think the style translation did a pretty good job from the perspective of a trained machine expert. This result gives us more confidence to use the style-translated images in our training set.

Up to this point, we have two possible approaches to utilize the newly generated images:

1. We translate PPAFM to experimental-like fake AFM images and use the identical atomic 3D structures of samples corresponding to PPAFM as the label to construct the training data. Then, we use this new training dataset to train a new structure discovery model as shown in Figure 5. Since the input training images have features similar to

real experimental images, the newly trained model should logically perform better on real AFM images, which is our goal.

2. When given experimental images, instead of directly inputting them to the structure discovery model, we first use the backward translation to turn the real AFM images into PPAFM-like images, then input the style-translated images to the model trained on only the simulation PPAFM dataset. Since the generated images have the feature of PPAFM images, the performance of the ML model on the generated fake PPAFM would be better than the performance when directly inputting real AFM images. In this approach, we only need to train the structure discovery model on simulation images, keeping the style translation as a separate prepossessing step before using the structure discovery model.

In the following discussion, I will only discuss the results obtained from the first approach: using fake AFM images to train a new structure discovery model and evaluating the performance using real experimental images.

3.2 Predictions of structure discovery model trained on fake AFM images

As shown in Figure 10, multi-layers of 2D experimental AFM images as one input sample are fed into the structure discovery model, and then the sample configuration in 3D is predicted. To evaluate the performance improvement of the new model trained on the fake AFM dataset, we use the original model trained on the simulation PPAFM as the reference.

The structure predictions of the samples from A to F are given by both original and new structure discovery models are shown in Figure 10. The two groups of predictions show some differences. In the original model predictions (Prediction v0 column), there are more isolated hydrogen atoms. In the new model predictions (Prediction v1 column), the occurrence of these isolated hydrogen atoms has been alleviated, indicating that the model trained on fake AFM images performs better or has greater potential for predicting real AFM images.

3.3 Performance evaluation workflow

However, for the experimental AFM images, we don't know the true sample configurations, i.e. we don't have the correct answers to compare to when we get the model predictions. Hence, it's difficult to evaluate the model performance. Therefore, we designed a performance evaluation workflow as shown in Figure 11. We have trained two structure discovery models with the only difference is that they are trained on different datasets. For both models, we've prepared three different kinds of input images: PPAFM images, fake AFM images, and real AFM images. Then two groups of predicted structures are obtained: One from the output of the original model v0: predictions 1, 2, and 3; another group from the output of the new model v1: predictions a, b, and c.

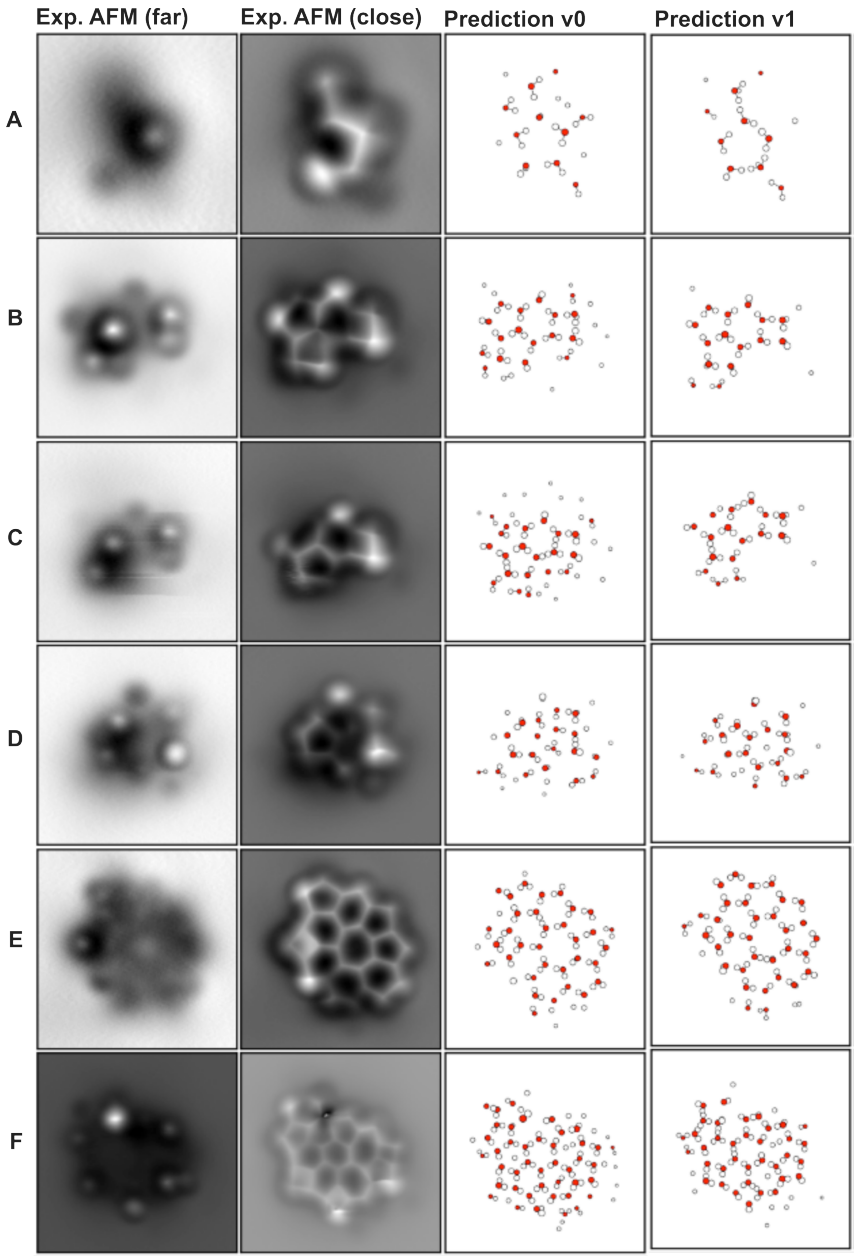

Figure 10: The inputs and predictions of structure discovery ML model. For one sample the inputs contain multiple layers of 2D AFM images from far to close. Predicion v0 column indicates the predictions given by the original model v0 trained on simulation PPAFM; Prediction v1 column contains the predictions given by the model v1 trained on fake AFM images.

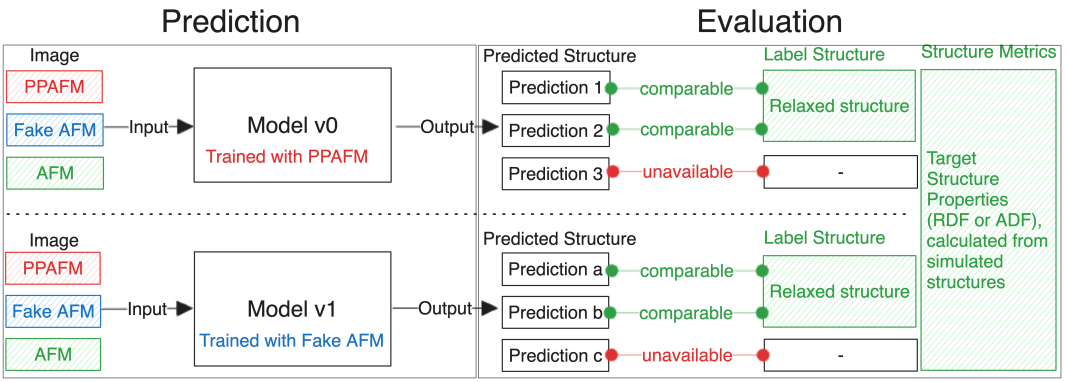

Figure 11: Performance evaluation workflow.

However, since the label (corrected) structures for experimental AFM images are not available, we cannot directly tell whether Prediction c is better than Prediction 3. Nevertheless, the true structures of all these water clusters should be close to their relaxed configuration because all the atoms prefer to stay at the most stable configuration in both DFT simulations and real physical experiments. Hence, instead of comparing the individual structure, we compare the structure properties which are calculated through many structures. Here, we use the radial distribution function (RDF) and angular distribution function (ADF) calculated from simulated relaxed structures. These structure properties should be logically very close to the real structure properties due to the accurate DFT calculations.

In the following discussion, I will be using the RDF and ADF functions of simulated relaxed sample structures as reference properties to assess the performance of the machine learning models. This will allow us to bypass the unknown true structure of individual samples.

3.4 Performance evaluation on experimental AFM images

The radial distribution function $g_{\alpha\beta}(r)$ measures the averaged particle β density as a function of distance from a central atom α and is calculated by

$$g_{\alpha\beta}(r) = \frac{n(r)}{4\pi r^2 \cdot \Delta r \cdot \rho} \tag{5}$$

where $n(r)$ is the atom β number between r and $r+\Delta r$, ρ is the number density of atom β. The angular distribution function $\mathrm{ADF}_{\alpha\beta\gamma}$ is calculated by counting the number of angles formed by angle $\widehat{\alpha\beta\gamma}$ within a radius cutoff.

Figure 12a shows an example of the relaxed water clusters on a gold surface. Structures like the example are used to generate the labels in the training data. Figures 12b, c show the RDF of O-O and O-H; Figures 12d, e show the ADF of H-O-H, O-H-O. These four structural properties from b to f are used as references to evaluate the performance of the structure discovery machine learning model. The radius cutoff for selecting the atoms around a center

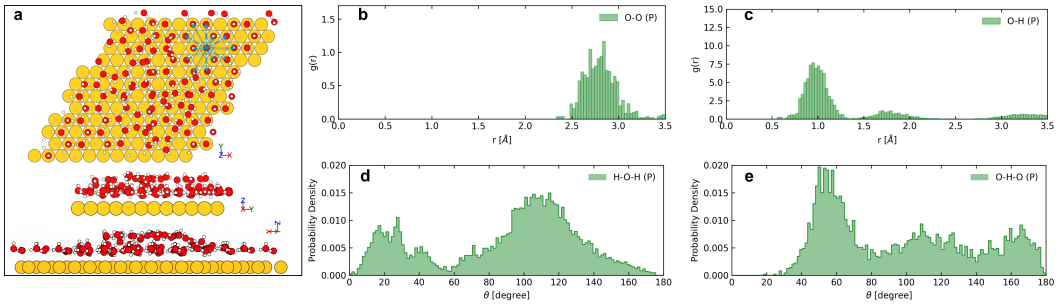

Figure 12: (a) One water clusters and Au surface configuration from three different perspectives; (b, c) The radial distribution function (RDF) for O-O and O-H pairs; and (d, e) the angular distribution functions (ADF) for H-O-H, and O-H-O angles of the relaxed structures used to generate PPAFM.

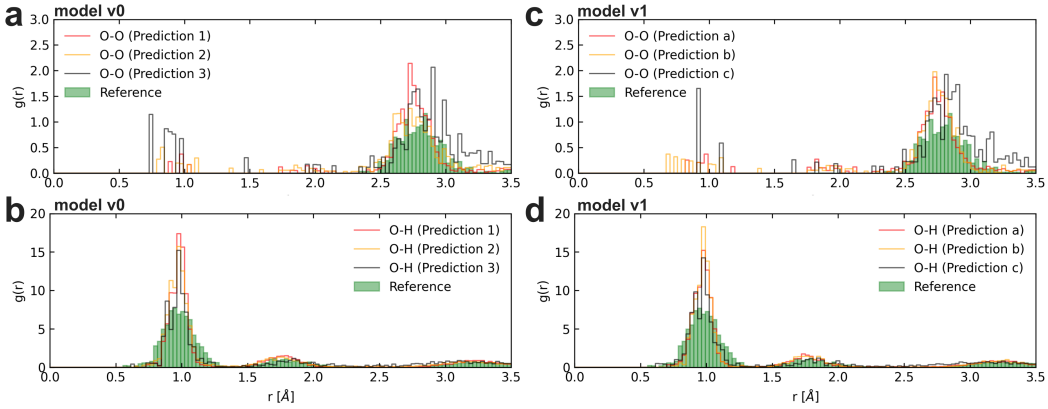

Figure 13: RDF comparison to the references for (a, b) Predictions 1–3 given by model v0 and (c, d) Predictions a–c given by model v1.

atom is set to be 3.5 Å, which covers the range of OH bonds of single water molecules, considers the hydrogen bond (H-bond) between water clusters, and is also greater than the average Au-Au distance in the substrate. Next for the structure predictions v0 and v1, we calculate these same properties. We then use the cosine similarity to quantitatively assess the deviation between the reference properties and properties calculated from predicted structures, which is defined as

$$S(\mathbf{X}_0, \mathbf{X}_i) = \frac{\mathbf{X}_0 \cdot \mathbf{X}_i}{\|\mathbf{X}_0\| \|\mathbf{X}_i\|} \tag{6}$$

where $S(\mathbf{X}_0, \mathbf{X}_i)$ quantifies the similarity between two high-dimensional vectors $\mathbf{X}$.

Figure 13 shows the RDF comparison to the reference properties, where (a, b) and (c, d) correspond to the original model v0 and the new structure discovery model v1. Three

types of predictions generate three curves in each plot, corresponding to three kinds of input data, namely, PPAFM, fake AFM, and real AFM. By comparing these three curves to the references and calculating the similarities to the reference values, we can estimate how the model performance changes when image distribution shifts from simulation to experiment. All the corresponding cosine similarities are listed in Table 1. Here are some observations:

1. The O-O RDF values in the range [0, 2] should be 0, i.e., no neighbor O atoms should appear in this short range, which is concluded from the reference O-O RDF. Hence, the non-zero values that appear in this range indicate wrong predictions.

2. Prediction 1 curve shown in Figure 13a, obtained from the output of the original model v0 trained on PPAFM images by input PPAFM images, shows that non-zero values appear in this forbidden range, indicating that this type of error happens when there is no image distribution shift.

3. As model input changes from PPAFM (Prediction 1) to real AFM (Prediction 3), this situation becomes worse, indicating that the original model cannot handle the real AFM images as well as it predicts simulation PPAFM images. Performance that becomes worse can also be revealed by the cosine similarity decreasing from 0.908 to 0.734.

4. In Figure 13c, we can see that model v1 trained on fake AFM still encounters the same issues as model v0. However, the cosine similarity value for Prediction c is 0.819, which is higher than the value of 0.734 for Prediction 3 in Figure 13a. This indicates that the new model performs better than the original model on experimental AFM images from the O-O RDF perspective.

5. In Figure 13b,d, the cosine similarity value (0.920) for Prediction c is higher than the value (0.881) for Prediction 3 in Figure 13a, indicating the new model's better performance on experimental AFM images from the O-H RDF perspective.

Similar to the RDF analysis, Figure 14 shows the ADF comparison to the reference properties, where (e, f) and (g, h) correspond to the original (v0) and new (v1)structure discovery model. Here are some observations:

1. When changing input images from PPAFM to real AFM, the ADFs are getting increasingly different from their references, indicating the performance becomes worse for both the original and new machine learning model.

2. As shown in Figure 14e,h, the cosine similarity value (0.921) for Prediction c is higher than the value (0.894) for Prediction 1, indicating that the new model performs better than the original model on experimental AFM images from the H-O-H ADF perspective.

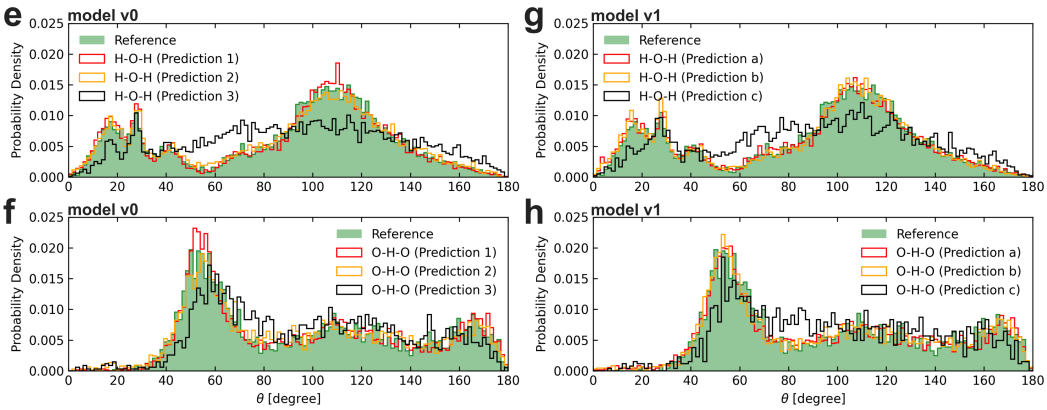

Figure 14: ADF comparison to the references for (e, f) original Predictions 1–3 given by model v0 and (g, h) new Predictions a–c given by model v1.

$S(\mathbf{X}_0, \mathbf{X}_i)$	Pred. 1	Pred. 2	Pred. 3	Pred. a	Pred. b	Pred. c
RDF O-O	0.908	0.918	0.734	0.907	0.907	0.819
RDF O-H	0.866	0.912	0.881	0.907	0.874	0.920
ADF H-O-H	0.989	0.990	0.894	0.990	0.989	0.921
ADF O-H-O	0.973	0.978	0.917	0.981	0.977	0.915

Table 1: Structure property cosine similarities between predictions and references. These values are viewed as indexes to evaluate the machine learning model performance from different perspectives for different kinds of input AFM images.

3. From the perspective of O-H-O ADF, the new model did not perform better than the original model on the experimental AFM images. Yet the corresponding similarity values for Prediction c and 3 are very close to each other.

From the RDF and ADF analysis, we found that the new model showed performance improvements in some structural metrics, including O-O RDF, O-H RDF, and H-O-H ADF, while no improvement was observed when evaluated by the O-H-O ADF metric. The O-H-O angles that contribute to the ADF curves are primarily those smaller than 120 degrees, which are not the angles forming the H-bonds. These angles are likely to occur when one of the O-H bonds in a water molecule points along the z-axis. This suggests that ADF curves mainly measure the angles aligned vertically along the z direction.

Since we did not treat multiple layers of 2D PPAFM as a whole unit and viewed 2D PPAFM images at different heights the same during CycleGAN training, converting a PPAFM image to an experimental-like AFM image may disrupt the consistency between the 2D AFM layers for a given sample, or cause lose of information in the z direction. This could explain why no performance improvement was observed for the O-H-O ADF metric.

4 Conclusions

In conclusion, we have applied a CycleGAN model to translate image styles between the simulation PPAFM and real AFM images and evaluated the style translation quality through a trained machine expert. As expected, the results show that the trained generator does have the ability to shift the PPAFM distribution towards the real AFM distribution.

We have used a water cluster structure discovery machine learning model as an example to test our ideas. We have constructed new training data by replacing the original PPAFM images with the style-translated fake AFM images and retrained a new model with this new dataset. By observing the predicted structure of water clusters on gold surfaces, we can observe to some extent performance improvement. However, since we do not have the structures for the experimental AFM images, we have designed a performance evaluation workflow based on structure properties including RDF and ADF to assess the model performance. We observed performance improvements in 3 out of the 4 criteria we tested. These results show the potential that the style translation-based method can assist us in better transferring machine learning from the simulation domain to the real experimental domain.

Looking ahead, more explorations will be tested from the following aspects: (i) More experimental images are planned in CycleGAN training; more experimental images mean more samples from the real AFM distribution, which can lead to better generalization. With broader and richer real AFM images, CycleGAN model will likely capture more realistic features from the real data, ultimately improving its style translation robustness. (ii) The effect of the weight λ of cycle consistency on the loss function of CycleGAN training. This parameter affects the style translation a lot. Now we have chosen to use relatively large λ to let the fake AFM images get high scores from the machine expert, and then use these fake AFM images to train the structure discovery model. But whether the bad fake AFM images would still help or even surpass the good fake AFM images in training the structure discovery model is a question. (iii) Backward translation as a separate processing step for the machine learning model trained on the simulation dataset. If the effectiveness of the idea is approved, we only need to focus on training highly accurate structure models using simulation datasets without worrying about the difference between simulation and experimental domains. Meanwhile, the AFM-to-PPAFM style translation generator acts as a bridge between the simulation and experimental domains, enabling the model to adapt effectively to real-world AFM images. This separation would simplify the workflow, allowing us to easily update either models as new data or experimental techniques become available, leading to continuous improvements without retraining the entire system.

References

[1] G. Binnig, C. F. Quate, and Ch. Gerber. Atomic force microscope. *Physical Review Letters*, 56(9):930–933, March 1986.

[2] Y. Martin, C. C. Williams, and H. K. Wickramasinghe. Atomic force microscope–force mapping and profiling on a sub 100-Å scale. *Journal of Applied Physics*, 61(10):4723–4729, May 1987.

[3] R García. Dynamic atomic force microscopy methods. *Surface Science Reports*, 47(6–8):197–301, September 2002.

[4] T. R. Albrecht, P. Grütter, D. Horne, and D. Rugar. Frequency modulation detection using high-q cantilevers for enhanced force microscope sensitivity. *Journal of Applied Physics*, 69(2):668–673, January 1991.

[5] Leo Gross, Fabian Mohn, Nikolaj Moll, Peter Liljeroth, and Gerhard Meyer. The chemical structure of a molecule resolved by atomic force microscopy. *Science*, 325(5944):1110–1114, August 2009.

[6] Prokop Hapala, Georgy Kichin, Christian Wagner, F. Stefan Tautz, Ruslan Temirov, and Pavel Jelínek. Mechanism of high-resolution STM/AFM imaging with functionalized tips. *Physical Review B*, 90(8), August 2014.

[7] Prokop Hapala, Ruslan Temirov, F. Stefan Tautz, and Pavel Jelínek. Origin of high-resolution IETS-STM images of organic molecules with functionalized tips. *Physical Review Letters*, 113(22), November 2014.

[8] Niko Oinonen, Aliaksandr V. Yakutovich, Aurelio Gallardo, Martin Ondráček, Prokop Hapala, and Ondřej Krejčí. Advancing scanning probe microscopy simulations: A decade of development in probe-particle models. *Computer Physics Communications*, 305:109341, December 2024.

[9] G. Kresse and J. Hafner. Ab initiomolecular dynamics for liquid metals. *Physical Review B*, 47(1):558–561, January 1993.

[10] G. Kresse and J. Furthmüller. Efficiency of ab-initio total energy calculations for metals and semiconductors using a plane-wave basis set. *Computational Materials Science*, 6(1):15–50, July 1996.

[11] G. Kresse and J. Furthmüller. Efficient iterative schemes forab initiototal-energy calculations using a plane-wave basis set. *Physical Review B*, 54(16):11169–11186, October 1996.

[12] Niko Oinonen, Lauri Kurki, Alexander Ilin, and Adam S. Foster. Molecule graph reconstruction from atomic force microscope images with machine learning. *MRS Bulletin*, 47(9):895–905, July 2022.

[13] Fabio Priante, Niko Oinonen, Ye Tian, Dong Guan, Chen Xu, Shuning Cai, Peter Liljeroth, Ying Jiang, and Adam S. Foster. Structure discovery in atomic force microscopy imaging of ice. *ACS Nano*, February 2024.

[14] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks, 2020.