Surfaces and Interfaces at the Nanoscale (SIN)

Aalto University School of Science

Enhancing AFM image analysis through machine learning with style translation and data augmentation

Jie Huang[a], **Niko Oinonen**[a, b], **Lauri Kurki**[a], **Adam S. Foster**[a, c]

[a] Department of Applied Physics, Aalto University, Finland

[b] Nanolayers Research Computing Ltd., London N12 0HL, United Kingdom

[c] WPI Nano Life Science Institute (WPI-NanoLSI), Kanazawa University, Japan

Contact Information:
Kide 2501, Department of Applied Physics
Aalto University, Helsinki, Finland

Email: jie.huang@aalto.fi

Abstract

Atomic Force Microscopy (AFM) is critical for atomic-scale nanostructure characterization. Simulations, especially using Particle Probe AFM (PPAFM), provide a cost-effective means for rapid image generation. Leveraging state-of-the-art machine learning models and substantial PPAFM-generated datasets, properties like molecular structures, electrostatic force potential, and molecular graphs can be accurately predicted using AFM images from simulations or experiments. However, transferring model performance from PPAFM to real AFM images poses challenges due to the subtle variations in real experimental data compared to the seemingly flawless nature of simulations. Our study explores Cycle GANs for style translation to augment data and improve the predictive accuracy of machine learning models in surface property analysis. Focused on mitigating the gap between simulated PPAFM and authentic AFM images, we optimize hyperparameters, showcasing the method's effectiveness through paired data comparisons. This research promises valuable insights, providing a novel approach to enhance machine learning model efficiency in the absence of abundant experimental data.

Materials and Methods

The Probe Particle Model (PPM) [1] provides a efficient simulation way to obtain AFM images. Combined with Machine Learning (ML), properties like molecular structures [2], electrostatic force potential [3] can be predicted by inputting simulated or experimental AFM images to an trained model. However, when we apply the trained model on the real experimental AFM images, the performance is not as good as it does in simulation images due to the difference between the PPAFM simulations and real experimental images. Images from these two domains are similar but not identical. What motivates us is to find a method that can use used to improve the performance of ML models in experimental data.

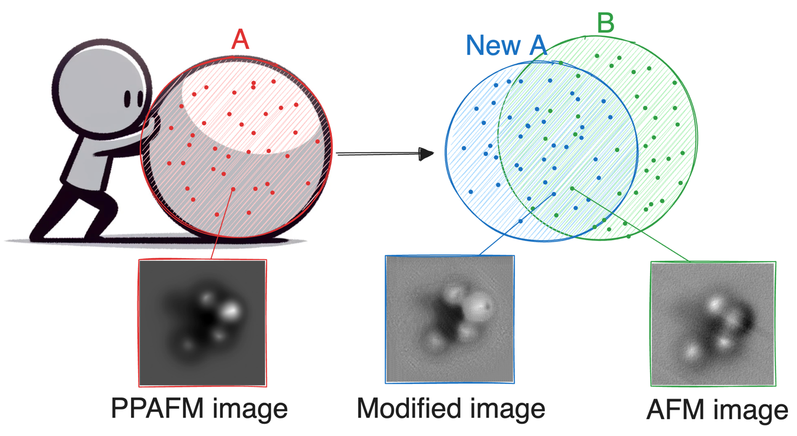

Figure 1: **The gap between the simulation and experimental** AFM images blocks the performance of machine learning models. One method is to **modify the simulation images** so that they **have the same trait or style as experimental images**, and add these modified images to the training set, i.e., **Data Augmentation**. If the newly trained model has better performance, the method is a good one.

We hypothesis the gap between the simulation and experimental AFM images blocks the performance of machine learning models. If some modifications are made on the simulation AFM images so that they have the same trait or style as experimental images, we can add the modified images to the training set when training machine learning models, i.e., Data Augmentation. If the newly trained model has better performance, this kind of modification is acceptable.

To generate a experiment-like image based on a given simulation PPAFM image, we need a generator which receives a simulation image as an input, and then output a experiment-like image. The generator can be a neural network based on convolutional neural networks. Here we use generative adversarial network (GAN). To train the generator a discriminator, another neural network, is a necessary part to encourage the generator to generate the images that looks like real. When it comes to our case, we have two sets of unpaired images come from simulation (domain A) and experimental images (domain B). We want to the simulation images can have the trait of experimental images. In other words, we want to do style translation from simulation domain to real experimental domain.

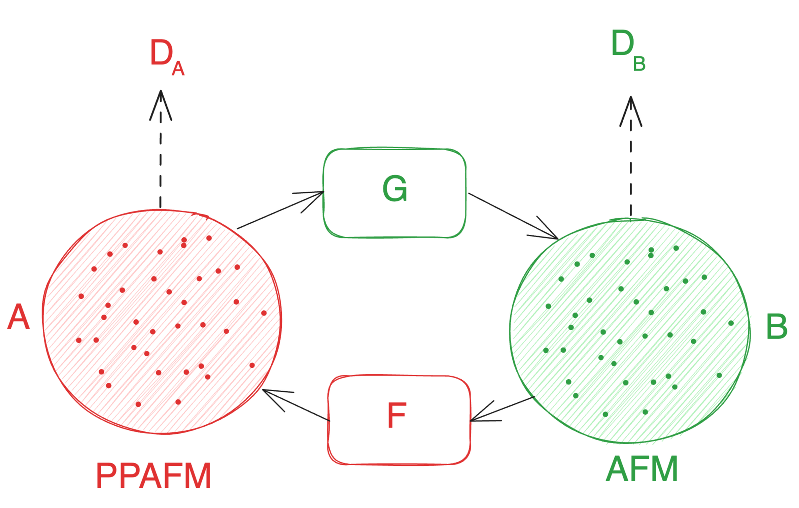

Figure 2: CycleGAN includes two mapping functions $G : A \rightarrow B$ and $F : B \rightarrow A$, and associated adversarial discriminators D_A and D_B encourages G to translate A into outputs indistinguishable from domain B, and vice versa for D_A and F.

The way we choose is CycleGAN [4], which uses the unpaired data to do image-to-image translation so that images in two domains interchange their style. Different from basic GAN which includes one generator and one discriminator, CycleGAN includes two generators (G and F) and two discriminators (D_B and D_A) as shown in Figure 2. Besides, another cycle consistency loss is added to train CycleGAN, which captures the intuition that if we translate from one domain to the other and back again we should arrive at where we started, and encourages $F(G(x)) \approx x$ and $G(F(y)) \approx y$, where x, y are images in domain A and domain B respectively. To train these neural networks G, F, D_A, D_B in the CycleGAN, a loss function is defined as follows:

$$\mathcal{L}(G, F, D_A, D_B) = \mathcal{L}_{\mathrm{GAN}}(G, D_B, A, B) + \mathcal{L}_{\mathrm{GAN}}(F, D_A, B, A) + \lambda \mathcal{L}_{\mathrm{cyc}}(G, F) \tag{1}$$

where

$$\mathcal{L}_{\mathrm{GAN}}(G, D_B, A, B) = \mathbb{E}_{x \sim P_B}[\log D_B(x)] + \mathbb{E}_{x \sim P_{FB}}[\log(1 - D_B(x))] \tag{2}$$

is the loss function for forward GAN, and similarly, $\mathcal{L}_{\mathrm{GAN}}(F, D_A, B, A)$ is the loss functions for backward GAN,

$$\mathcal{L}_{\mathrm{cyc}}(G, F) = \mathbb{E}_{x \sim P_A}[\|F(G(x)) - x\|_1] + \mathbb{E}_{y \sim P_B}[\|G(F(y)) - y\|_1] \tag{3}$$

is the cycle consistency loss and λ is a hyperparameter. Larger λ leads to more conservative generators. To train these neural networks, two datasets in PPAFM domain and experiment domain need to be prepared. We've used 610 PPAFM simulation images based on the configurations of water molecules on Au surface. And 609 experimental AFM images including water on Au and Cu surfaces are used as the training set of experimental domain.

Results

With the suitable hyperparameters include λ, network structures of generators and discriminators, the trained generator can be used as a tool to add some experimental features to the PPAFM simulation images as shown in Figure 3. Some background shadow and noises are added to the PPAFM images while the outline are still the same.

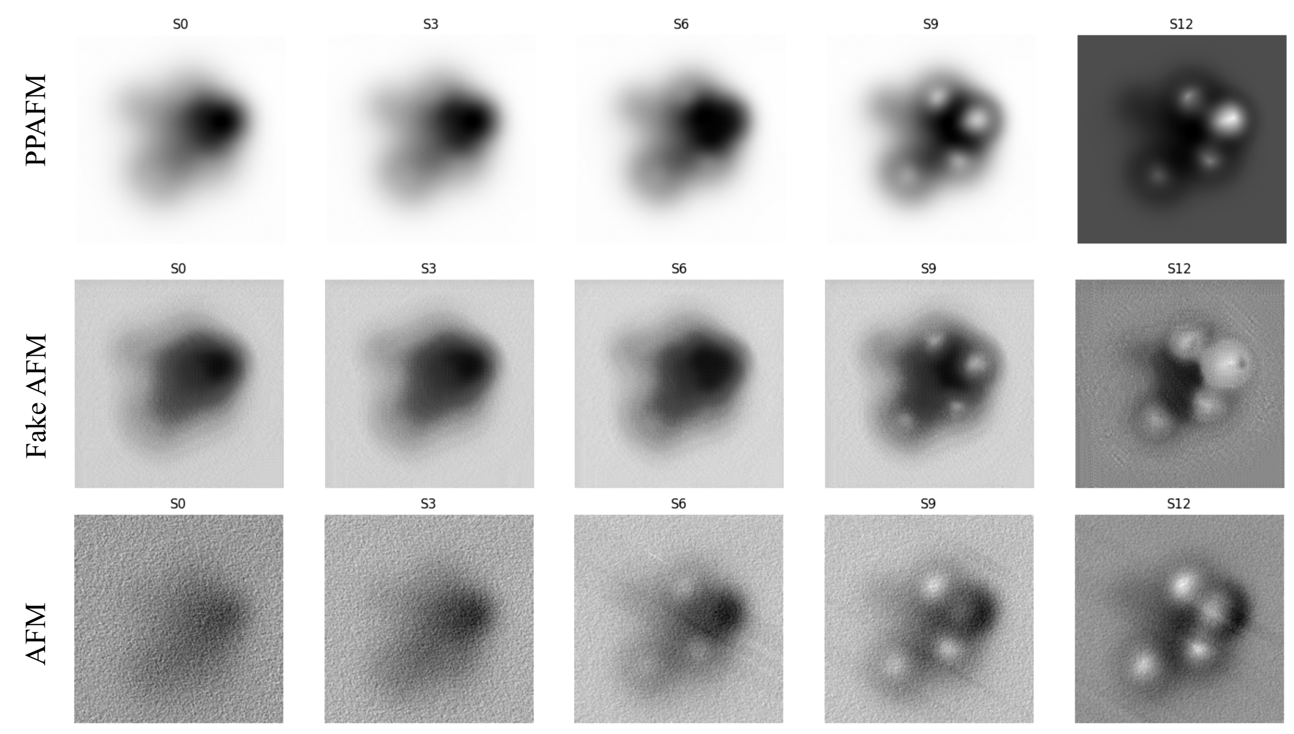

Figure 3: We use paired images for water molecules on Au surface at different heights for better comparison. First row indicates simulation images generated by PPAFM, and the second row include the the corresponding generated AFM images, and the last row shows the real AFM images at diffenret height.

It is worth noting that the hyperparameters and dataset would affect the style translation a lot. A proper way to select the 'good' style-transformed images are needed.

Conclusions and Future Works

- Using CycleGAN, the trained generator can learn some features of experimental AFM images based on the PPAFM images.
- An automatic way to evaluate the quality of generated images need to be designed.
- Generated images should be added to the training set of some machine learning models.

References

[1] Prokop Hapala, Ruslan Temirov, F. Stefan Tautz, and Pavel Jelínek. Origin of high-resolution iets-stm images of organic molecules with functionalized tips. *Phys. Rev. Lett.*, 113:226101, Nov 2014.

[2] Benjamin Alldritt, Prokop Hapala, Niko Oinonen, Fedor Urtev, Ondrej Krejci, Filippo Federici Canova, Juho Kannala, Fabian Schulz, Peter Liljeroth, and Adam S. Foster. Automated structure discovery in atomic force microscopy. *Science Advances*, 6(9), February 2020.

[3] Niko Oinonen, Chen Xu, Benjamin Alldritt, Filippo Federici Canova, Fedor Urtev, Shuning Cai, Ondřej Krejčí, Juho Kannala, Peter Liljeroth, and Adam S. Foster. Electrostatic discovery atomic force microscopy. *ACS Nano*, 16(1):89–97, 2022. PMID: 34806866.

[4] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks, 2017.

Acknowledgements

The calculations presented above were performed using computer resources within the Aalto University School of Science "Science-IT" project.