Denoising Scanning Tunneling Microscopy Images with Machine Learning

Frédéric Joucken¹, John L. Davenport², Zhehao Ge², Eberth A. Quezada-Lopez², Takashi Taniguchi³, Kenji Watanabe⁴, Jairo Velasco Jr.², Jérôme Lagoute⁵, and Robert A. Kaindl¹

¹Department of Physics, Arizona State University, Tempe, AZ 85287, USA
²Department of Physics, University of California, Santa Cruz, CA 95064, USA
³International Center for Materials Nanoarchitectonics, National Institute for Materials Science, 1-1 Namiki, Tsukuba 305-0044, Japan
⁴Research Center for Functional Materials, National Institute for Materials Science, 1-1 Namiki, Tsukuba 305-0044, Japan
⁵Laboratoire Matériaux et Phénomènes Quantiques, UMR 7162, Université Paris Diderot – Paris 7, Sorbonne Paris Cité, CNRS, UMR 7162 case courrier 7021, 75205 Paris 13, France

Abstract. Machine learning (ML) methods are extraordinarily successful at denoising photographic images. The application of such denoising methods to scientific images is, however, often complicated by the difficulty in experimentally obtaining a suitable expected result as an input to training the ML network. Here, we propose and demonstrate a simulation-based approach to address this challenge for denoising atomic-scale scanning tunneling microscopy (STM) images, which consists of training a convolutional neural network on STM images simulated based on a tight-binding electronic structure model. As model materials, we consider graphite and its mono- and few-layer counterpart, graphene. With the goal of applying it to any experimental STM image obtained on graphitic systems, the network was trained on a set of simulated images with varying characteristics such as tip height, sample bias, atomic-scale defects, and non-linear background. Denoising of both simulated and experimental images with this approach is compared to that of commonly-used filters, revealing a superior outcome of the ML method in the removal of noise as well as scanning artifacts – including on features not simulated in the training set. An extension to larger STM images is further discussed along with intrinsic limitations arising from training set biases. The approach demonstrated here provides an effective way to remove noise and artifacts from typical STM images, yielding the basis for further feature discernment and automated processing.
1. INTRODUCTION

Since its first demonstration by Binnig and Rohrer,\(^1\) scanning tunneling microscopy (STM) has become a powerful and essential tool in materials science due to its ability to measure and image electronic structure with sub-angstrom resolution.\(^2\) Although the present paper focuses on imaging, STM is used for many other purposes, including atomic-scale electronic spectroscopy, atom manipulation, or determination of momentum-resolved electronic band structure.\(^3\)–\(^5\) The recent rise of machine learning (ML) has spawned completely new approaches for analyzing complex scientific data.\(^6\)–\(^9\) First applications of ML to STM-related problems have also been reported, enabling automated identification and classification of atomic-scale patterns, materials phases, and defects\(^10\)–\(^14\) as well as advances towards autonomous STM operation through ML-based tip shaping and lithography.\(^13\),\(^15\),\(^16\)

The problem of reducing noise in STM images – including both statistical noise and artifacts - has, however, not yet been addressed. In the processing of STM images for analysis, the reduction of such noise provides several benefits. First, the enhancement of the signal-to-noise ratio can improve the level of confidence when interpreting experimental STM data. Secondly, STM measurements are inherently slow due to their scanning nature, and many acquired images are discarded by the experimentalists because of limited quality achieved. Thus, an efficient ML-based filter could increase the yield of STM experiments and the amount of data available for analysis. Finally, it can be expected that denoising will enhance the performance of other subsequent computational or ML-based steps such as defect recognition, autonomous data acquisition, and control.

Supervised ML models based on CNNs have been incredibly successful at denoising photographic images.\(^17\)–\(^27\) The common strategy for achieving high performance denoising of photographic images is to train the network on images onto which noise (typically Gaussian) is added intentionally. The corrupted images are then used as the inputs of the CNN while the original
images are used as the expected result, also referred to as “ground-truth” (GT) in the models. However, this strategy cannot be implemented straightforwardly for denoising scientific images because the expected result of scientific images is more difficult to assess. For some experimental methods, it is possible to produce such fundamental images by simply acquiring images for a long time.\textsuperscript{28,29} This assumes that both statistical noise and artifacts diminish with increased scanning. However, in STM imaging even images with low statistical noise can be far from the expected GT representation because of the role played by the tip in the imaging process. Indeed, the density of states of the tip is convolved with the density of states of the sample to produce the image.\textsuperscript{30} Even if efforts are generally put in acquiring images with tips that have a constant or known density of states so that their effect on the imaging is understood,\textsuperscript{31–35} it is always difficult to assess whether an STM image can be considered as the true representation of the underlying electronic structure.

As a guide, experimental STM results are often compared to simulated ones to assess their validity. The simplest way to simulate an STM image is to completely neglect the effect of the tip and assume that the image is proportional to the local density of states (LDOS) of the sample.\textsuperscript{30} Here, we exploit this idea by training a CNN on intentionally corrupted images simulated using the tight-binding approach for computing the LDOS. The tight-binding method allows for producing a large training set in a reasonable amount of time, contrary to more computationally demanding electronic structure computation techniques such as density functional theory. The trained CNN is then used to denoise experimental STM images, as illustrated in Fig. 1(a). We compare this approach to other denoising techniques, and we discuss the advantages and inconveniences offered by our simulation-based denoising technique for STM.

Simulation-based denoising was recently introduced by Mohan \textit{et al.} for the case of transmission electron microscopy (TEM) images.\textsuperscript{36} However, both the nature of the electronic interaction underlying the TEM imaging process as well as the generation of the training set images differs fundamentally from that required for STM denoising. Moreover, we aim at training a model that can be applied to STM images of an entire family of materials encompassing both graphite and its
two-dimensional counterparts of mono- and multi-layer graphene. As discussed below, this requires the generation of a very diverse training set.

Another approach which has naturally been popular with scientific denoising is unsupervised learning, where the need for a GT image as the expected result is removed altogether.\textsuperscript{37–42} It has been successfully applied to fluorescence microscopy\textsuperscript{37–39,42} and TEM data.\textsuperscript{41} Although unsupervised approaches are appealing for denoising STM images, they are challenging to adapt to STM for two reasons. First, obtaining a large training set is difficult in STM because the time needed to acquire an image is relatively long compared to other microscopy techniques. Secondly, some assumptions on the nature of the noise must be made to build the loss function in unsupervised learning.\textsuperscript{40} Although some of the noise in STM images can be assumed to take a certain statistical form,\textsuperscript{43} the influence of the tip shape and density of states cannot be accounted for by assuming statistical noise.

II. COMPUTATIONAL MODEL

For the CNN, the model is implemented in Python with the Keras\textsuperscript{44} software library and its TensorFlow backend.\textsuperscript{45} The architecture is built on the U-Net type of networks,\textsuperscript{46} which was shown by Mohan \textit{et al.} to be most efficient for denoising corrupted simulated images.\textsuperscript{36} Our adapted version of the U-Net architecture is depicted in Fig. 1(b). It consists of two ‘down’ building blocks between which the size of the arrays is divided by two by the Keras \texttt{maxpooling} operation, a lower building block, and two ‘up’ building blocks between which the size of the arrays is multiplied by two by the \texttt{upsampling} operation. Each block has two convolution layers with $x_F$ filters rectified linear unit (ReLU) activation functions that introduce the non-linearity of the network, followed by \texttt{batchnormalization} which centers the mean output of each block close to 0 and its standard deviation close to 1. The size of the 2D convolution kernel is set to $3 \times 3$ and the number of filters in the first block of each layer was $n = 32$ and is doubled or halved, respectively, after each \texttt{maxpooling} or \texttt{upsampling} operation. The last layer has a tanh
activation function to best adapt the overall output range to that of the GT images (so-called labels). As the loss function, i.e. quantity that is minimized during training, we choose the mean absolute error between the outputs and labels. To optimize the network, we used the Adam algorithm in Keras, which considers the exponentially-weighted average of the loss function gradient.\textsuperscript{47} The learning rate that sets the step size of the gradient descent algorithm was set to 0.001 per iteration. A typical training run was done on $3 \times 10^4$ simulated images with an additional 500 images used for validation at each iteration. Figure 1(c) shows the evolution of the loss function with the iteration (epoch) number during training, along with the loss computed on the validation images.

**Figure 1. Overview of our machine learning (ML) denoising approach applied to STM images.** (a) Workflow of the ML denoising based on simulated STM images. The convolutional neural network (CNN) is trained on a set of simulated STM images which are intentionally corrupted, and then applied to experimental images. Scale bars of 0.5 nm. (b) Architecture of the CNN, which includes two encoding layers, one bottom layer, and two decoding layers with details given in the text. (c) Evolution of the loss function i.e. the mean absolute error between the output
and corresponding pristine images, after each iteration (epoch) during training, along with the equivalent loss function obtained on a validation images set.

For the generation of the training set images we used the pybinding package, a Python package for tight-binding calculations, for the calculation of the energy- and spatially-dependent electronic structure of finite graphene nanosheets.\textsuperscript{48} In this approach, the kernel polynomial method\textsuperscript{49} is used to compute the projected density of states \(c_{i,E}\) on each atomic site \(i\) at energy \(E\). The STM tunneling current at a given sample bias \(V_{\text{Sample}} = E/e\) (\(e\) is the electron charge) is then computed for \(V > 0\) as
\[
I(x, y, z, V) \propto \sum_i \left( \sum_{E'=0}^{E} c_{i,E'} \right) \left| \pi_{z,i}(x, y, z) \right|^2,
\]
where \(\pi_{z,i}(x, y, z) \propto z e^{-r/\lambda}\) represents the value at \((x, y, z)\) of the \(\pi_z\) orbital located at site \(i\). Here, \(r = \sqrt{(x - x_i) + (y - y_i)^2 + (z - z_i)^2}\) defines the distance between the tip location and the carbon atom at site \(i\). In the above, \(\lambda \approx n a_0/Z = 0.018\) nm for the given principal quantum number \(n = 2\), atomic number \(Z = 6\), and Bohr radius \(a_0\). When the bias voltage \(V_{\text{Sample}} < 0\), i.e. for tunneling from the sample to the tip, the sum over the projected density of states in STM current calculation above is replaced by \(\sum_{E'=E}^{0} c_{i,E'}\). For computational efficiency, the STM training images were calculated for a constant height \(z\) throughout each image. The lateral image size is chosen randomly between 1 and 4 nm with \(64^2\) pixels.

In training and applying the CNN, we restrict ourselves to a particular material since it would be impractical to simulate STM images for all known materials. We chose monolayer and Bernal-stacked bilayer graphene as model materials. It should be noted that, because STM is surface sensitive, our model can in principle be applied to any graphitic material.

To make our denoising CNN as broadly applicable as possible, we train it on images that are as diverse as possible. To that end, we introduce the following variations, randomly chosen for each image: (i) Both monolayer and Bernal-stacked bilayer graphene are simulated. (ii) Dopants, simulated by changing the onsite energy to 10 eV at the position of the dopant, were randomly distributed. (iii) A hyperbolic background was added to the images. (iv) The lattice orientation, tip height \(z\), and the sample bias between -0.5 to 0.5 V were randomly set. Moreover, two types of
noise are included: (v) a Gaussian height noise with randomly chosen standard deviation $d_z$ that is added onto each pixel, and (vi) a random horizontal spatial offset $dx$ is added for each line of the image, simulating a common line noise effect seen in STM images. The training images were first normalized to between -0.5 and 0.5, and the standard deviation $d_z$ of the added Gaussian height noise limited to a maximum value of 0.07 on this scale. In turn, the maximum value for the randomly chosen line noise $dx$ was 0.1 nm. A set of representative simulated STM images with (‘corrupted’) and without noise (‘GT’) used for training are shown in Supplemental Materials Fig. S1. Our code for producing the simulated images and for the ML model is provided on GitHub.\textsuperscript{50}

The production of simulated images and training of the CNN model were carried out on a modern computer with graphics processing unit in several hours, while denoising experimental images once the model is trained occurs on the order of seconds.

III. RESULTS AND DISCUSSION

A. Denoising of Simulated Images

Before its application to experimental data, we first analyze the performance of the ML-based model on simulated STM data and compare it to other denoising approaches. The third column in Fig. 2(a) shows the results of applying the trained model on two simulated corrupted images, not seen during CNN training. The fourth and fifth columns in Fig. 2(a) show the images after applying instead a Gaussian smoothing algorithm or a singular value decomposition (SVD) filter, respectively. The Gaussian smoothing consists of a convolution between the image and a 2D Gaussian profile with a fixed width $\sigma$. We chose $\sigma = 1.1$ pixels by maximizing the peak signal-to-noise ratio (PSNR) as defined below as a function of $\sigma$ (see Supplementary Materials). In turn, SVD-based denoising uses the approach by Somnath et al.\textsuperscript{51} It consists of tiling the $N \times N$ pixels image into $m \times m$ pixels windows, flattening the resulting 2D tile arrays onto 1D vectors and stacking them. A SVD decomposition is then applied to the resulting $(N-m+1)^2 \times m^2$ matrix. The filtering consists of keeping only the first $t$ principal components. We set $m = 8$ and $t = 8$
for all SVD filtering presented in this work. Visual inspection in Fig. 2(a) of the results of these different denoising approaches reveals that the ML model performs well compared to the other methods. Both the height noise and line noise are almost entirely suppressed by the ML model, which is not the case for the other approaches. The contrast of the atomic lattice is also higher in the ML case.

Figure 2. Denoising of simulated STM images with different methods. (a) The first column displays simulated STM images of pristine monolayer graphene and of bilayer graphene with a N dopant (black arrow). The second column displays the corresponding corrupted images. The third, fourth, and fifth columns show the corresponding ML-, Gaussian-, and SVD-denoised images, respectively. The parameters for monolayer graphene are: \( V_{\text{Sample}} = +0.23 \text{ eV}, z = 0.98 \text{ nm}, \) height noise \( d_z = 0.08 \), line noise \( dx = 0.103 \text{ nm} \), and for bilayer graphene: \( V_{\text{Sample}} = -0.49 \text{ eV}, z = 0.68 \text{ nm}, dz = 0.041, dx = 0.107 \text{ nm} \). Scale bars are 1 nm. (b) Comparison between the peak signal-to-noise ratio (PSNR) for the various approaches as a function of the simulated noise level (see text).
Figure 2(b) plots the PSNR as a function of the image noise level, for each type of denoising approach. The PSNR between a noise-free image and its corrupted counterpart is often used to evaluate the performance of denoising schemes and is defined as $PSNR = 20 \log(l_{max}) - 10 \log(MSE)$, where $l_{max}$ is the maximum pixel value of the image (1 in our case) and the mean squared error (MSE) is the average squared difference between the pixels of the noise free and the corrupted images. For producing the graph shown in Fig. 2(b) we generated ten sets of 200 corrupted images, with each set corresponding to a different value of the height noise $dz$ and line noise $dx \equiv dz$ plotted on the horizontal axis. Indeed, Fig. 2(b) quantitatively confirms the visual impression that the ML approach performs better than the Gaussian smoothing and SVD methods at high noise levels. Importantly, even for low noise levels, ML denoising retains the PSNR while the other methods reduce the image quality for given fixed parameters. In the case of the ML approach, this consistent performance results from the equivalence of minimizing the loss function (mean absolute error) during training and maximizing the PSNR.

**B. Experimental STM Image Denoising**

We now show how our ML denoising method works on experimental data, first on images with features that were previously included in the training set. Figure 3 shows the experimental STM data and the results of different denoising approaches. The experimental STM images were obtained from scans carried out at liquid helium temperatures (4-5 K) without an applied external magnetic field. The images of pristine monolayer and bilayer graphene were obtained on graphene devices (see elsewhere for details on the making of these devices\(^5\)). The images of doped and defected graphene were obtained on graphene grown on the carbon face of silicon carbide that was intentionally doped with nitrogen.\(^5\) The experimental STM images were acquired at constant tunneling current, as customary to avoid crashing the tip, and thus represent the corresponding tip height variation. While the ML model is trained on constant-height data, the two modes generally produce similar images.
Figure 3: Denoising of experimental STM image features simulated in the training. Experimental STM images of (a) pristine monolayer graphene with $V_{\text{sample}} = -0.5$ V, (b) bilayer graphene with $V_{\text{sample}} = +0.3$ V and (c) monolayer graphene including an N dopant as indicated by the arrow ($V_{\text{sample}} = +0.5$ V). Scale bars are 0.5 nm. Results of applying the ML-based denoising to the experimental STM images are displayed in the second column. The results of Gaussian smoothing and SVD-based filtering are reported in the third and fourth columns, respectively.

The first column of Figs. 3(a) and 3(b) corresponds to experimentally measured STM images of pristine monolayer graphene (MLG) and bilayer graphene (BLG), respectively, and in Fig. 3(c) to monolayer graphene with a dopant (indicated by an arrow). Experiments and density functional theory calculations have shown that for single nitrogen substitution in graphene, the highest electronic density is found on the three carbon atoms nearest neighbors to the nitrogen dopant,\textsuperscript{53,54} hence the three lobes clearly visible in Fig. 3(c). The second column, in Fig. 3 shows the images denoised with our ML model. For comparison, the third and fourth columns show the same experimental images denoised with the Gaussian smoothing and the SVD filter, respectively.
Because an uncorrupted image is inherently unavailable in the experimental case, a quantitative evaluation of the relative performance in terms of the PSNR is not possible. However, as in the case of denoising of the simulated STM images discussed above, a visual inspection reveals that the ML approach performs well compared to the other filters. In particular, we note that the ML model more effectively removes the noise that appears as horizontal lines in the experimental images. The contrast corresponding to the atomic lattice also appears higher for the ML denoising [see Fig. 3(c)].

In Fig. 4, we show how our ML denoising performs on features that did not appear in the simulations used during training, where only single dopants were randomly incorporated. The first column displays three experimental images. The top image [Fig. 4(a)] is a 3N pyridine defect in graphene, which consists of a carbon vacancy accompanied by the replacement of its nearest neighbors with three nitrogen atoms in a pyridine configuration. The center image [Fig. 4(b)] is the so-called “flower” defect in graphene, commonly observed in graphitic systems. The bottom image [Fig. 4(c)] displays a grain boundary between two graphene domains, often found in graphene/SiC(0001) samples. Visual inspection indicates a good performance of ML-based denoising (second column) when compared to the other two approaches (third and fourth columns). As in the cases discussed above, the contrast of the atomic lattice appears sharper for the ML-based method. That is especially true for the grain boundary defect [Fig. 4(c)], where the Gauss and SVD filters do not reproduce the graphene lattice that is visible on both sides of the grain boundary in the experimental and the ML-denoised images.

A clear advantage of the ML approach over the other methods is the absence of parameters to be tuned once the model is trained. By comparison, other denoising methods including the Gaussian and SVD techniques entail parameters that need to be readjusted to different values depending on the type of image. In the images in Figs. 3(c) and 4(c), the same parameter $\sigma = 1.1$ pixel was used for Gaussian smoothing. This parameter works well for the dopant in Fig. 3(c), but it is not adapted for the grain boundary defect [Fig. 4(c)] and its associated high protrusion, as evidenced by the
poorly resolved graphene lattice in the Gaussian-smoothed image. The same remark holds for the SVD approach, where two parameters must be set. The same parameters ($m = 8$ and $t = 8$, see above) were used in Figs. 3 and 4. The denoising effectiveness varies widely as evident from comparing the SVD-denoising of images in these figures.

Figure 4: Denoising of experimental monolayer graphene images with features not simulated in the training set. (a) $3N$ pyridine defect ($V_{\text{sample}} = +0.5$ V). (b) ‘Flower’ defect ($V_{\text{sample}} = +0.5$ V). (c) Grain boundary between two graphene domains ($V_{\text{sample}} = +0.4$ V). Scale bars are 1 nm. Results of applying the ML-based denoising to the experimental images are displayed in the second column. The results of Gaussian smoothing and SVD filtering are reported in the third and fourth columns, respectively. The dashed black boxes in (a) and (b) encompass a donut forming the donut-like charge density wave feature (see text) which is partially smeared out by the ML-filtering.
Figure 5: ML-based denoising of extended images larger than the training images. (a) 10 × 10 nm² image (256² pixels) showing monolayer graphene with N dopants. (b) ML-denoised image obtained by subdividing panel (a) into 4 × 4 images, applying the ML model to these images, and patching the ML-denoised images back together.

C. Extended STM Images

Because the training set of our ML model was limited to 64×64 pixels images and a lateral side length between 1–4 nm, it cannot be directly applied to larger images. While it is possible in principle to train the network on wider images with a higher pixel count, this rapidly increases the computational resources needed both for generation of the training set and for the training of the model. An alternative solution to apply ML denoising to larger images is to subdivide the large image into smaller images, apply the network to the smaller images, and then stitch the small images back together. This approach is illustrated in Fig. 5, on a 10 × 10 nm² image containing 256² pixels. The image in Fig. 5(a) is subdivided into 16 images of 64² pixels (2.5 × 2.5 nm²) onto
which the ML model is applied and then assembled back together. The resulting denoised image is shown in Fig. 5(b). Except for faint lines at the borders between the sub-images, the resulting large ML-processed image (cf. insets in Fig. 5) is visually cleaner than the experimental image.

D. Influence of the Training Set and Limitations

We now discuss how the inclusion of dopants, background, quasiparticle interference, and other features in the simulated STM training images affects the outcome of ML-based denoising. In Fig. 6, we show the same experimental images as in Fig. 4 and compare how the performance of the ML-denoising is affected by the inclusion or omission of features in the image set used for training the model. The case I (second column) was trained on simulated images containing only pristine monolayer and bilayer graphene. Perhaps unsurprisingly, this model poorly handles the filtering of images containing defects. The case II (third column) was trained on images of pristine monolayer and bilayer graphene, but with inclusion of quasiparticle interferences (QPI), but no dopant. Quasiparticle interferences in general are induced by charge-carrier scattering from defects such as dopants, step edges, or grain boundaries, and they produce standing-wave patterns.\textsuperscript{59,60} For the simulated images in the training set, QPI is inherently included in the tight-binding calculation due to scattering off the boundaries of the simulated graphene nanosheet. In this case, it becomes prominent only when nanosheet size is small (cf. Supplementary Fig. S3). Such simulated images with evident QPI can be seen e.g. in the zig-zag type patterns in Fig. 2(a) [top left image] and in Figs. S1(a) and S1(j). Surprisingly, training with such QPI-pattern images alone is sufficient for the model to perform relatively well on denoising of images that include defects, as evident from Fig. 6. Only the case of the grain boundary [Fig. 6(c)] is poorly handled by this approach. The case III in Fig. 6 combines images containing dopants in the training set and results in the model being able to handle more complex defects such as the grain boundary.
Figure 6: Influence of the introduction of various features in the simulated images used for training the ML model. (a)-(c) Experimental images as in Fig. 4. Scale bars are 1 nm. Case I: results from the ML-filter model trained only with simulated pristine monolayer and bilayer graphene STM images. Case II: results from the ML model trained with an image set that includes simulated images displaying quasiparticle interference patterns (see text and Suppl. Materials) but no dopant. Case III: results from the ML model trained with images containing all the features (same as Fig. 4, second column; see section II for details on all the included features).

Finally, we discuss a feature that is partially smeared out by our ML-filtering. As evidenced in Fig. 4, a donut-like pattern is present, highlighted by the dotted box in Figs. 4(a) and 4(b). This donut-like pattern arises from a charge-density wave due to a Kékulé distortion whose periodicity is \( \sqrt{3} \) larger than that of the graphene lattice. Similar STM signatures have been observed recently as a broken symmetry state in charge neutral graphene, in intense magnetic fields,\textsuperscript{61,62} and can be reproduced with a tight-binding model if one assumes that valleys and sublattices are locked i.e. that charge carriers in one of the two inequivalent valleys of graphene comprise only carriers on one of the two sublattices.\textsuperscript{61} Similar Kékulé distortions have been observed by STM, in the absence
of magnetic field, in Li-intercalated graphene\textsuperscript{63} and in graphene grown epitaxially on copper.\textsuperscript{64} However, these previously reported STM signatures do not match the distinct donut-like pattern that we report here for defected graphene, which corresponds to a certain mixture of K and K’ states.\textsuperscript{61}

The tight-binding model underlying the simulated STM images used to train our model does not reproduce the donut-like charge density wave state, since we had no a priori reason to assume any valley-sublattice locking. This lack of inclusion in the training set explains why the donut-like pattern tends to be partially smeared out by the ML-based denoising model as evident from the second column images in Figs. 4(a) and 4(b). This issue illustrates well an intrinsic limitation from the bias induced by a non-representative training set in machine learning.\textsuperscript{9,65–67} While the feature is not completely smeared out [cf. Fig. 4(b)] this demonstrates that care should be taken when applying ML-based models to experimental data in physics experiments if previously unexplored questions are tackled.

**IV. CONCLUSIONS**

In summary, we have introduced a machine learning model for denoising STM images of graphitic systems which is trained on simulated images. The wide variety of simulated images in the training set allows us to apply the model to a broad range of experimental images. We also demonstrate that by tiling the model can be applied to denoise images that are larger than those used in training while retaining a rapid computational speed. The relevance of including different features in the simulated training set was analyzed, revealing that the inclusion of QPI patterns alone results in surprisingly good ML denoising. Finally, limitations of ML-based denoising were illustrated in the case of an unexpected charge-density wave distortion. The ML denoising approach presented here is expected to be an effective tool for facilitating the visual analysis of STM images and for enhancing subsequent machine learning techniques applied to defect detection and classification, atom manipulation, and scan automation.
ACKNOWLEDGMENTS. J.V.J. acknowledges support from the National Science Foundation under Grant No. DMR-1753367 and the Army Research Office under Contract No. W911NF-17-1-0473. K.W. and T.T. acknowledge support from the Elemental Strategy Initiative conducted by the MEXT, Japan, Grant No. JPMXP0112101001 and JSPS KAKENHI Grant No. JP20H00354.

REFERENCES

1. Binnig, G., Rohrer, H., Gerber, Ch. & Weibel, E. Surface Studies by Scanning Tunneling Microscopy. Physical Review Letters 49, 57–61 (1982).
2. C. Julian Chen. Introduction to Scanning Tunneling Microscopy. (Oxford University Press, USA, 2021).
3. Crommie, M. F., Lutz, C. P. & Eigler, D. M. Confinement of Electrons to Quantum Corrals on a Metal Surface. Science (1979) 262, 218–220 (1993).
4. Crommie, M. F., Lutz, C. P. & Eigler, D. M. Imaging standing waves in a two-dimensional electron gas. Nature 363, 524–527 (1993).
5. Fischer, Ø., Kugler, M., Maggio-Aprile, I., Berthod, C. & Renner, C. Scanning tunneling spectroscopy of high-temperature superconductors. Reviews of Modern Physics 79, 353–419 (2007).
6. Bedolla, E., Padierna, L. C. & Castañeda-Priego, R. Machine learning for condensed matter physics. Journal of Physics: Condensed Matter 33, 053001 (2020).
7. Pederson, R., Kalita, B. & Burke, K. Machine learning and density functional theory. Nature Reviews Physics 4, 357–358 (2022).
8. Carleo, G. et al. Machine learning and the physical sciences. Reviews of Modern Physics 91, 45002 (2019).
9. Yano, J. et al. The case for data science in experimental chemistry: examples and recommendations. Nature Reviews Chemistry 6, 357–370 (2022).
10. Ziatdinov, M., Wong, C. Y. & Kalinin, S. v. Finding simplicity: unsupervised discovery of features, patterns, and order parameters via shift-invariant variational autoencoders. (2021) doi:10.48550/arxiv.2106.12472.
11. Zhang, Y. et al. Machine learning in electronic-quantum-matter imaging experiments. Nature 570, 484–490 (2019).
12. Roccapriore, K. M. et al. Revealing the Chemical Bonding in Adatom Arrays via Machine Learning of Hyperspectral Scanning Tunneling Spectroscopy Data. ACS Nano 15, 11806–11816 (2021).
13. Rashidi, M. et al. Deep learning-guided surface characterization for autonomous hydrogen lithography. *Machine Learning: Science and Technology* **1**, 025001 (2020).

14. Ziatdinov, M., Fuchs, U., Owen, J. H. G., Randall, J. N. & Kalinin, S. v. Robust multi-scale multi-feature deep learning for atomic and defect identification in Scanning Tunneling Microscopy on H-Si(100) 2x1 surface. (2020) doi:10.48550/arxiv.2002.04716.

15. Rashidi, M. & Wolkow, R. A. Autonomous Scanning Probe Microscopy in Situ Tip Conditioning through Machine Learning. *ACS Nano* **12**, 5185–5189 (2018).

16. Krull, A., Hirsch, P., Rother, C., Schiffrin, A. & Krull, C. Artificial-intelligence-driven scanning probe microscopy. *Communications Physics* **3**, 54 (2020).

17. Zhang, K., Zuo, W. & Zhang, L. FFDNet: Toward a fast and flexible solution for CNN-based image denoising. *IEEE Transactions on Image Processing* **27**, 4608–4622 (2018).

18. Liu, P., Zhang, H., Zhang, K., Lin, L. & Zuo, W. Multi-level wavelet-CNN for image restoration. in Proceedings of the IEEE conference on computer vision and pattern recognition workshops 773–782 (2018).

19. Jia, X., Liu, S., Feng, X. & Zhang, L. Focnet: A fractional optimal control network for image denoising. in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 6054–6063 (2019).

20. Lefkimmiatis, S. Universal denoising networks: a novel CNN architecture for image denoising. in Proceedings of the IEEE conference on computer vision and pattern recognition 3204–3213 (2018).

21. Kligvasser, I., Shaham, T. R. & Michaeli, T. xunit: Learning a spatial activation function for efficient image restoration. in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2433–2442 (2018).

22. Lefkimmiatis, S. Non-local color image denoising with convolutional neural networks. in Proceedings of the IEEE conference on computer vision and pattern recognition 3587–3596 (2017).

23. Jain, V. & Seung, S. Natural image denoising with convolutional networks. *Adv Neural Inf Process Syst* **21**, (2008).

24. Tai, Y., Yang, J., Liu, X. & Xu, C. Memnet: A persistent memory network for image restoration. in Proceedings of the IEEE international conference on computer vision 4539–4547 (2017).

25. Zhang, K., Zuo, W., Chen, Y., Meng, D. & Zhang, L. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE transactions on image processing* **26**, 3142–3155 (2017).

26. Mao, X., Shen, C. & Yang, Y.-B. Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. *Adv Neural Inf Process Syst* **29**, 2802–2810 (2016).

27. Chen, Y. & Pock, T. Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration. *IEEE Trans Pattern Anal Mach Intell* **39**, 1256–1272 (2016).
28. Manifold, B., Thomas, E., Francis, A. T., Hill, A. H. & Fu, D. Denoising of stimulated Raman scattering microscopy images via deep learning. *Biomedical Optics Express* **10**, 3860–3874 (2019).

29. Kim, Y. et al. Deep learning-based statistical noise reduction for multidimensional spectral data. *Review of Scientific Instruments* **92**, 073901 (2021).

30. Tersoff, J. & Hamann, D. R. Theory and Application for the Scanning Tunneling Microscope. *Physical Review Letters* **50**, 1998–2001 (1983).

31. Zhang, Y. et al. Giant phonon-induced conductance in scanning tunnelling spectroscopy of gate-tunable graphene. *Nature Physics* **4**, 627–630 (2008).

32. Ernst, S., Wirth, S., Rams, M., Dolocan, V. & Steglich, F. Tip preparation for usage in an ultra-low temperature UHV scanning tunneling microscope. *Science and Technology of Advanced Materials* **8**, 347–351 (2007).

33. Rodrigo, J. G. & Vieira, S. STM study of multiband superconductivity in NbSe2 using a superconducting tip. *Physica C: Superconductivity* **404**, 306–310 (2004).

34. Ekvall, I., Wahlström, E., Claesson, D., Olin, H. & Olsson, E. Preparation and characterization of electrochemically etched W tips for STM. *Measurement Science and Technology* **10**, 11–18 (1999).

35. Stocker, M., Pfeifer, H. & Koslowski, B. Calibration of tip and sample temperature of a scanning tunneling microscope using a superconductive sample. *Journal of Vacuum Science & Technology A* **32**, 031605 (2014).

36. Mohan, S. et al. Deep denoising for scientific discovery: a case study in electron microscopy. *arXiv preprint arXiv:2010.12970* (2020).

37. Zhang, Y. et al. A Poisson-Gaussian Denoising Dataset With Real Fluorescence Microscopy Images. in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2019).

38. Prakash, M., Lalit, M., Tomancak, P., Krull, A. & Jug, F. Fully Unsupervised Probabilistic Noise2Void. in *2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI)* 154–158 (2020). doi:10.1109/ISBI45749.2020.9098612.

39. Khademi, W., Rao, S., Minnerath, C., Hagen, G. & Ventura, J. Self-Supervised Poisson-Gaussian Denoising. in *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)* 2130–2138 (2021). doi:10.1109/WACV48630.2021.00218.

40. Laine, S., Karras, T., Lehtinen, J. & Aila, T. High-Quality Self-Supervised Deep Image Denoising. in *Advances in Neural Information Processing Systems* (eds. Wallach, H. et al.) vol. 32 (Curran Associates, Inc., 2019).

41. Krull, A., Buchholz, T.-O. & Jug, F. Noise2Void - Learning Denoising from Single Noisy Images. (2018) doi:10.48550/arxiv.1811.10980.

42. Batson, J. & Royer, L. Noise2Self: Blind Denoising by Self-Supervision. (2019) doi:10.48550/arxiv.1901.11365.
43. Ge, J.-F., Ovadia, M. & Hoffman, J. E. Achieving low noise in scanning tunneling spectroscopy. *Review of Scientific Instruments* **90**, 101401 (2019).

44. Chollet, F. & Others. Keras. https://github.com/fchollet/keras.

45. Martín~Abadi *et al*. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. (2015).

46. Ronneberger, O., Fischer, P. & Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015* (eds. Navab, N., Hornegger, J., Wells, W. M. & Frangi, A. F.) 234–241 (Springer International Publishing, 2015).

47. Kingma, D. P. & Ba, J. Adam: A Method for Stochastic Optimization. (2014) doi:10.48550/ARXIV.1412.6980.

48. Moldovan, D., Andelković, M. & Peeters, F. pybinding v0.9.5: a Python package for tight-binding calculations. (2020) doi:10.5281/ZENODO.4010216.

49. Weiße, A., Wellein, G., Alvermann, A. & Fehske, H. The kernel polynomial method. *Reviews of Modern Physics* **78**, 275–306 (2006).

50. https://github.com/fjoucken/Denoise_STM.

51. Somnath, S. *et al*. Feature extraction via similarity search: application to atom finding and denoising in electron and scanning probe microscopy imaging. *Advanced Structural and Chemical Imaging* **4**, 3 (2018).

52. Joucken, F. *et al*. Direct Visualization of Native Defects in Graphite and Their Effect on the Electronic Properties of Bernal-Stacked Bilayer Graphene. *Nano Letters* **21**, 7100 (2021).

53. Joucken, F. *et al*. Localized state and charge transfer in nitrogen-doped graphene. *Physical Review B - Condensed Matter and Materials Physics* **85**, 161408 (2012).

54. Zhao, L. *et al*. Visualizing Individual Nitrogen Dopants in Monolayer Graphene. *Science (1979)* **333**, 999–1003 (2011).

55. Tison, Y. *et al*. Electronic interaction between nitrogen atoms in doped graphene. *ACS Nano* **9**, 670–678 (2015).

56. Cockayne, E. *et al*. Grain boundary loops in graphene. *Physical Review B* **83**, 195425 (2011).

57. Varchon, F., Mallet, P., Magaud, L. & Veuillen, J.-Y. Rotational disorder in few-layer graphene films on 6H-SiC(000-1): A scanning tunneling microscopy study. *Physical Review B* **77**, 165415 (2008).

58. Tison, Y. *et al*. Grain Boundaries in Graphene on SiC(0001) Substrate. *Nano Letters* **14**, 6382–6386 (2014).

59. Crommie, M. F., Lutz, C. P. & Eigler, D. M. Imaging standing waves in a two-dimensional electron gas. *Nature* **363**, 524–527 (1993).
60. Hoffman, J. E. Spectroscopic scanning tunneling microscopy insights into Fe-based superconductors. *Reports on Progress in Physics* **74**, 124513 (2011).

61. Xiaomeng, L. *et al.* Visualizing broken symmetry and topological defects in a quantum Hall ferromagnet. *Science (1979)* **375**, 321–326 (2022).

62. Coissard, A. *et al.* Imaging tunable quantum Hall broken-symmetry orders in graphene. *Nature* **605**, 51–56 (2022).

63. Bao, C. *et al.* Experimental Evidence of Chiral Symmetry Breaking in Kekulé-Ordered Graphene. *Physical Review Letters* **126**, 206804 (2021).

64. Gutiérrez, C. *et al.* Imaging chiral symmetry breaking from Kekulé bond order in graphene. *Nature Physics* **12**, 950–958 (2016).

65. Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K. & Galstyan, A. A Survey on Bias and Fairness in Machine Learning. *ACM Comput. Surv.* **54**, (2021).

66. Karniadakis, G. E. *et al.* Physics-informed machine learning. *Nature Reviews Physics* **3**, 422–440 (2021).

67. Torralba, A. & Efros, A. A. Unbiased look at dataset bias. in *CVPR 2011* 1521–1528 (2011). doi:10.1109/CVPR.2011.5995347.
Supplemental Materials for:
Denoising Scanning Tunneling Microscopy Images with Machine Learning

Frédéric Joucken\textsuperscript{1}, John L. Davenport\textsuperscript{2}, Zhehao Ge\textsuperscript{2}, Eberth A. Quezada-Lopez\textsuperscript{2}, Takashi Taniguchi\textsuperscript{3}, Kenji Watanabe\textsuperscript{4}, Jairo Velasco Jr.\textsuperscript{2}, Jérôme Lagoute\textsuperscript{5}, and Robert A. Kaindl\textsuperscript{1}

\textsuperscript{1}Department of Physics, Arizona State University, Tempe, AZ 85287, USA
\textsuperscript{2}Department of Physics, University of California, Santa Cruz, CA 95064, USA
\textsuperscript{3}International Center for Materials Nanoarchitectonics, National Institute for Materials Science, 1-1 Namiki, Tsukuba 305-0044, Japan
\textsuperscript{4}Research Center for Functional Materials, National Institute for Materials Science, 1-1 Namiki, Tsukuba 305-0044, Japan
\textsuperscript{5}Laboratoire Matériaux et Phénomènes Quantiques, UMR 7162, Université Paris Diderot – Paris 7, Sorbonne Paris Cité, CNRS, UMR 7162 case courrier 7021, 75205 Paris 13, France

Figure S1 shows representative simulated STM images used for training. The corresponding most important parameters are indicated below each image: whether it is monolayer graphene or bilayer graphene (MLG or BLG), the lateral size of the image, the energy $E$, the tip height $z$ (arb. u.), and the two noise parameters $dz$ and $dx$. $dz$ is the standard deviation of the Gaussian noise added on the image (in arb. u.) and $dx$ is the upper limit for the randomly chosen lateral offset between each horizontal lines of the image (in nm). Randomly distributed dopants are also visible on some images.

In Fig. S2, the evolution of the peak signal-to-noise ratio (PSNR) is shown as a function of the standard deviation of the Gaussian filter (in pixels) applied to corrupted simulated images. The PSNR has been computed for 200 images and the error bars correspond to the standard deviation of the ensemble of the 200 images around the average. The maximum PSNR is observed around 1.1, hence our choice of $\sigma = 1.1$ for the Gaussian filter used for the data presented in the main text.
Figure S3 illustrates the method to induce quasiparticle interference (QPI) patterns in the simulated STM image. To introduce QPI patterns, it suffices to consider graphene nanosheets which are only slightly larger than the area considered to produce the STM image (located at the center of the graphene nanosheet). The QPI are then produced by the scattering of the quasiparticle off the edges of the simulated nanosheet.\textsuperscript{1,2} In Fig. S3(a), the nanosheet is 2 nm larger than the STM image (4 nm vs. 2 nm). In Fig. S3(b), the nanosheet is 10 nm larger than the STM image (12 nm vs. 2 nm). The other parameters are the same. One can see that QPI patterns are prominent in Fig. S3(a), whereas almost no QPI pattern is visible in Fig. S3(b).
Figure S1. Representative sample of simulated STM images used for training the model. Below each image, the main parameters are indicated: whether it is monolayer graphene or bilayer graphene (MLG or BLG), the lateral size of the image, the energy E, the tip height z, and the two noise parameters dz and dx. dz is the standard deviation of the Gaussian noise added on the image (in arb. u.) and dx is the upper limit for the randomly chosen lateral offset between each horizontal lines of the image (in nm). Randomly distributed dopants are also visible on some images.
**Figure S2:** Evolution of the peak signal-to-noise ratio (PSNR) with the standard deviation of a Gaussian filter (in pixels). The PSNR has been computed for 200 images and the error bars correspond to the standard deviation of the ensemble of the 200 images around the average.
Figure S3: Illustration of the method to induce quasiparticle interference (QPI) patterns in the simulated STM image. To introduce QPI patterns, it suffices to consider graphene nanosheet which are only slightly larger than the area considered to produce the STM image (located at the center of the graphene nanosheet). (a) The nanosheet is 2 nm larger than the STM image (4 nm vs. 2 nm). (b) The nanosheet is 10 nm larger than the STM image (12 nm vs. 2 nm). The other parameters are the same. One can see that QPI patterns are prominent in panel (a), whereas almost no QPI pattern is visible in panel (b).

References

1. Mahmood, A., Mallet, P. & Veuillen, J.-Y. Quasiparticle scattering off phase boundaries in epitaxial graphene. *Nanotechnology* **23**, 055706 (2012).

2. Tesch, J. *et al.* Impurity scattering and size quantization effects in a single graphene nanoflake. *Physical Review B* **95**, 75429 (2017).