Title

Generalization Across Experimental Parameters in Neural Network Analysis of High-Resolution Transmission Electron Microscopy Datasets

Permalink

https://escholarship.org/uc/item/3s0431n8

Journal

Microscopy and Microanalysis, 30(1)

ISSN

1431-9276

Authors

Sytwu, Katherine

DaCosta, Luis Rangel

Scott, Mary C

Publication Date

2024-03-07

DOI

10.1093/micmic/ozae001

Generalization Across Experimental Parameters in Neural Network Analysis of High-Resolution Transmission Electron Microscopy Datasets

Katherine Sytwu, Luis Rangel DaCosta, Mary C Scott

Microscopy and Microanalysis, 2024, **30**, 85–95
https://doi.org/10.1093/micmic/ozae001
Advance access publication 29 January 2024
Original Article

Microscopy AND Microanalysis

Generalization Across Experimental Parameters in Neural Network Analysis of High-Resolution Transmission Electron Microscopy Datasets

Katherine Sytwu[1,*], Luis Rangel DaCosta[1,2], and Mary C. Scott[1,2,*]

[1]*Molecular Foundry, Lawrence Berkeley National Laboratory, 1 Cyclotron Road, Berkeley, CA 94720, USA*
[2]*Materials Science and Engineering, University of California Berkeley, 2607 Hearst Ave, Berkeley, CA 94720, USA*

***Corresponding authors:** Katherine Sytwu, E-mail: ksytwu@lbl.gov; Mary C. Scott, E-mail: MCScott@lbl.gov

Abstract

Neural networks are promising tools for high-throughput and accurate transmission electron microscopy (TEM) analysis of nanomaterials, but are known to generalize poorly on data that is "out-of-distribution" from their training data. Given the limited set of image features typically seen in high-resolution TEM imaging, it is unclear which images are considered out-of-distribution from others. Here, we investigate how the choice of metadata features in the training dataset influences neural network performance, focusing on the example task of nanoparticle segmentation. We train and validate neural networks across curated, experimentally collected high-resolution TEM image datasets of nanoparticles under various imaging and material parameters, including magnification, dosage, nanoparticle diameter, and nanoparticle material. Overall, we find that our neural networks are not robust across microscope parameters, but do generalize across certain sample parameters. Additionally, data preprocessing can have unintended consequences on neural network generalization. Our results highlight the need to understand how dataset features affect deployment of data-driven algorithms.

Key words: generalization, machine learning, nanoparticles, transmission electron microscopy

Introduction

With increasing amounts of data from faster detector speeds and new automated microscope setups, there is a pressing need for high-throughput analysis of high-resolution transmission electron microscope (HRTEM) images of nanomaterials. HRTEM enables atomic-scale visualization of material structure with high temporal resolution, making it a useful imaging modality for high-throughput and *in situ* transmission electron microscope (TEM) experiments of nanoparticle systems and behavior. The most promising HRTEM image analysis methods to date have been based on convolutional neural networks (CNNs), a class of machine-learning models that naturally take advantage of spatial correlations in image data (Madsen et al., 2018) and have outperformed traditional image processing methods at common microscopy analysis tasks like denoising and nanoparticle segmentation (Groschner et al., 2021; Vincent et al., 2021). These algorithms utilize a framework in which patterns and trends are learned from a large corpus of data, called the training set, and then evaluated on data the algorithm has not seen during training. The subsequent performance then depends on various choices made during model training, including the network architecture which sets the mathematical framework of the unknown model parameters, the loss function which constructs the overall optimization problem, and the training data which influence the learned model features.

While CNNs can achieve high performance, CNNs and other machine-learning models have also been empirically shown to not generalize, i.e., they do not perform well on data that differ from the data used during model training (Recht et al., 2019; Torralba & Efros, 2011). This inability to generalize across different datasets has consequences for deploying CNNs for large-scale microscopy analysis, for instance in determining which networks are reusable across multiple experiments or reliable for data streams with changing conditions, like *in situ* data. Generalization issues are typically categorized in two ways: 1) in-distribution generalization, or the ability to generalize on data that has been nominally sampled from a similar distribution as the training data and whose drop in performance is commonly referred to as the "generalization gap" and 2) out-of-distribution generalization, or the ability to extrapolate to new data that is known to be different from the training set. While there has been a growing amount of research focused on algorithmic solutions to minimize generalization issues (Shen et al., 2021), we first need to understand under what conditions generalization problems occur. Such an analysis requires domain-specific knowledge which associates model performance gaps with domain-knowledge of the modified image or data features (Liu et al., 2020; Kaufmann & Vecchio, 2021; Li et al., 2023).

With HRTEM data, it is unclear what types of images are considered out-of-distribution from others. While metadata information like sample and/or imaging parameters may designate images as different from one another, it is unknown whether a trained neural network would be sensitive to such changes given the limited number of image features typically seen in HRTEM images. For example, should we expect a single neural network to perform well across a wide range of

Received: June 20, 2023. **Revised:** December 19, 2023. **Accepted:** January 6, 2024

nanoparticle sizes? And, if not, under what conditions will we expect the performance of the model to decrease? There is limited knowledge on how the training dataset affects neural network performance, despite our (often) relatively complete understanding of both the sample and imaging process. While there have been some attempts to understand the effect of the training dataset with simulated data (Mohan et al., 2022), we lack experimental benchmarks to fully validate these generalization effects. With more data-driven models being proposed and developed by the microscopy community, there is a need to understand the reusability of these models on new datasets, and under what conditions they succeed or fail (Larsen et al., 2023; Wei et al., 2023).

In this paper, we systematically investigate the robustness of neural networks trained to identify nanoparticles in HRTEM images by benchmarking neural network performance on datasets that experimentally differ from the training set (Fig. 1a). We examine the effect of microscope and sample parameters in the training set, including magnification, electron dosage, nanoparticle diameter, and nanoparticle material. As an example task, we focus on segmentation, or pixelwise classification, of atomically resolved crystalline nanoparticles against an amorphous background, a typical initial image processing step for further analysis of atomic defects or crystal structure (Groschner et al., 2021), or nanoparticle dynamic behavior (Yao et al., 2020). By training and evaluating neural networks on experimental HRTEM datasets curated with controlled imaging and sample parameters, we not only qualitatively identify conditions under which we expect networks to generalize (or not) but also provide new datasets with extensive metadata that enable benchmarking HRTEM image analysis methods under specified microscopy conditions. In addition to our observations on training set effects, we demonstrate how data preprocessing influences generalization, providing a case study in preparing and utilizing experimental TEM data for machine-learning methods.

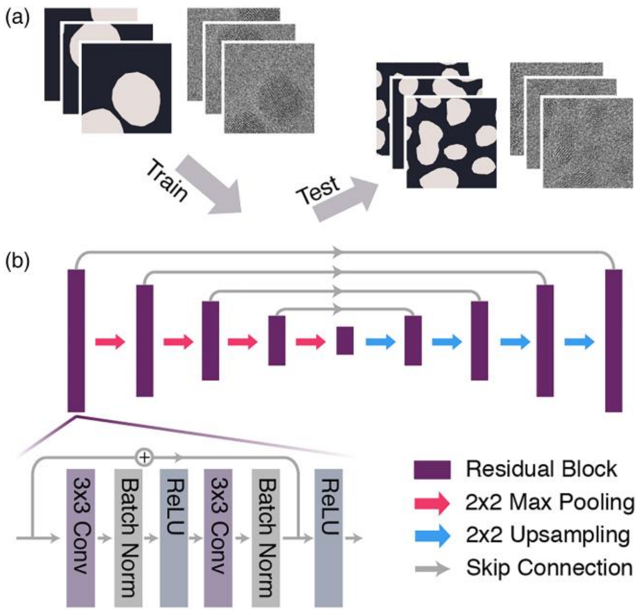

Fig. 1. Overview of the network training and testing protocols. (**a**) Datasets with specified metadata parameters are labeled and created from large HRTEM images and used to train and test neural networks. (**b**) The residual UNet neural network architecture used for all models in this paper, consisting of four residual "encoding" blocks and four residual "decoding" blocks, all connected by either max pooling downsampling layers or upsampling layers. Each residual block consists of a series of convolution, batch norm, and ReLU layers with a residual connection adding the input features to the transformed output of the block before the final nonlinear activation.

Materials and Methods

Sample Preparation

Au nanoparticles of size 2.2 nm with citrate ligands were purchased from Nanopartz; 5, 10, and 20 nm Au nanoparticles capped with tannic acid were purchased from Ted Pella; 5 nm Ag nanoparticles with citrate ligands were purchased from nanoComposix; 5 nm CdSe nanoparticles with oleylamine ligands were purchased from Strem Chemicals. To create a TEM sample from aqueous nanoparticle solutions (Au, Ag), an ultrathin carbon grid (Ted Pella) was plasma cleaned with a shield for 3 s to promote hydrophilicity, then $5\,\mu$L of the purchased nanoparticle solution was dropcast onto the grid, let sit for 5 min, and excess liquid was wiped off with a Kimwipe. For the CdSe nanoparticles, the nanoparticle solution was diluted to 0.625% of the original concentration with hexane, and $5\,\mu$L of the diluted nanoparticle solution was dropcast onto an ultrathin carbon grid (Ted Pella) and let evaporate.

TEM Imaging

HRTEM images were taken with a TEAM 0.5 aberration-corrected microscope operated at 300 kV and a OneView camera (Gatan) at full resolution (4, 096 × 4, 096 pixels). Microscope magnification was either set to 160, 205, 260, or 330 kX magnification (corresponding to 0.042, 0.033, 0.026, and 0.02 nm/pixel) and the electron dosage was set between 80 and 884 e/Å^2. Note that dosage values are not quantitatively accurate due to an uncalibrated FluCam screen, but all values are off by the same factor. HRTEM images were taken at Gaussian focus of the substrate, as approximated by a human operator via a combination of examining the live image and its corresponding fast Fourier transform, which leads to images that are close to Gaussian focus of the nanoparticle.

Preprocessing and Dataset Creation

All HRTEM images were labeled by hand into segmented images using Labelbox, an online image labeling platform (Labelbox, 2023). To create a dataset, raw images (and their corresponding labels) were selected from the larger data repository using metadata (i.e., microscope conditions, nanoparticle parameters, etc.), and then preprocessed into a dataset (Sytwu et al., 2023). Preprocessing consisted of four steps: 1) removal of X-ray artifacts, 2) flat-field correction, 3) image value rescaling, and 4) divide into smaller patches. We apply all preprocessing steps by image to ensure that our methods scale with new data (i.e., adding more images to a dataset) and are reflective of model deployment, which is likely to be done by image. X-ray artifacts were removed by averaging the surrounding pixels of outlier points above a certain threshold (1,500 counts) above the median counts. For flat-field correction, we estimate the uneven illumination using iterative weighted linear regression to a 2D Bezier basis ($n = 2$, $m = 2$; Sadre et al., 2021), and divide out the estimated illumination profile. The iterative reweighting lessens the contribution from nanoparticle regions such that the substrate regions are primarily used to determine the uneven illumination. The pixel values of each image, X, are then rescaled using either normalization:

$$X_{\text{norm}} = \frac{X - \min(X)}{\max(X) - \min(X)} \tag{1}$$

standardization:

$$X_{\text{standard}} = \frac{X - \mu_X}{\sigma_X} \tag{2}$$

where μ_X is the image mean and σ_X is the image SD, or a percentile-based scaling procedure similar to Digital Micrograph:

$$X_{\text{percentile}} = \begin{cases} 0 & \text{if } X_i < p_{0.1} \\ \dfrac{X_i - p_{0.1}}{p_{99.9} - p_{0.1}} & \text{if } p_{0.1} < X_i < p_{99.9} \\ 1 & \text{if } X_i > p_{99.9} \end{cases} \tag{3}$$

where $p_{0.1}$ is the 0.1 percentile image value and $p_{99.9}$ is the 99.9 percentile image value. Finally, images are divided into 512 × 512 pixel patches to reduce GPU memory requirements during network training and patches that are mostly substrate are removed to obtain better class balance. Image patches are converted to and stored as single-precision floating point arrays in the preprocessing pipeline, which exceeds the original bit-depth of the OneView camera and is thus effectively lossless. The datasets used in this paper are described in more detail in Supplementary Table S1.

Neural Network Training and Testing

Neural networks are trained on a single dataset from Supplementary Table S1, chosen via selections upon experimental metadata, e.g., only image patches of 5 nm Au nanoparticles at 330 kX magnification; at no point are datasets mixed during training. To train a neural network, we need to first split up a given dataset of image patches into training, validation, and test subsets. The training set is used during model training to help determine the learned features, the validation set is used to determine the best model hyperparameters (i.e., how long to train), and the test set evaluates model performance. To account for potential variations due to the choice of test set, we evaluate model performance using fivefold cross validation. In fivefold cross validation, the dataset is split into five folds, and we train five independent networks where each network uses a different fold as a test set, and the rest of the patches are used in the training or validation set. Patches are assigned sequentially to the training, validation, and test sets such that it is less likely for patches from similar image regions to end up in both the training and test sets. In total, the dataset is split such that 70% of the dataset is used as the training set, 10% used as the validation set, and 20% used as the test set. Data augmentation is often used to synthetically increase dataset size, whether to increase the number of images seen during training or to test on a wider variety of images during testing. Here, we increase our training, validation, and test set sizes by 8× by including the eight dihedral image transformations (all unique combinations of 90° rotations, horizontal flips, and vertical flips), taking advantage of the rotational invariance of top-down perspective images. We then shuffle each training, validation, and test set into a random order.

Our neural network model architecture is a residual variant of the UNet architecture (Ronneberger et al., 2015), a common neural network architecture used for image segmentation, which consists of four encoder and mirrored decoder blocks that are connected by a bridge and skip connections. The UNet architecture (and its variants) has been previously successful at segmenting HRTEM images of nanoparticles (Groschner et al., 2021; Sytwu et al., 2022; Larsen et al., 2023). In this residual variant, batch normalization layers and residual connections are added to each encoder and decoder block as described by He et al. (2016) and depicted in Figure 1b; these residual connections have been generally shown to improve performance and optimization stability (He et al., 2016). We also reduce the number of filters (and thus the number of training parameters) to prevent overfitting during training. Neural networks are trained under a supervised learning framework, using a cross-entropy loss function with a learning rate of 1e-4 with an Adam optimizer (default parameters). During training, we additionally augment 50% of the images with random rotations between 0 and 360° which empirically produces smoother prediction edges and further synthetically increases our training set size; we do not apply these random rotations when measuring performance on the validation or test sets. During training, data is fed to the neural network in batches; we keep our batch size to 32 patches. We train for 250 epochs and save the neural network model with the lowest validation loss within those 250 epochs. With our specified image patch size, batch size, and neural network architecture hyperparameters, training utilizes approximately 10 Gb of GPU memory. All training is done locally on a NVIDIA RTX3090 GPU; with our specified epoch length, training a single neural network takes between 20 min and 1 h, depending on the size of the training dataset. We note that our hyperparameters are partially chosen to lead to model convergence within a reasonable amount of time, given the large number of networks we need to train. Hyperparameter tuning (see Supplementary Material) leads to models with similar if not slightly better performance, but much longer training times (up to and on order of 4 h).

We evaluate our models using the hard dice score, also known as the F1 score, which quantifies the similarity between the prediction and expert-provided label. The hard dice score can be calculated by $\frac{2\text{TP}}{2\text{TP}+\text{FP}+\text{FN}}$ for a binary classification, where TP is the number of true positive pixels, FP is the number of false positive pixels, and FN is the number of false negative pixels. The hard dice score ranges between 0 (for complete disagreement) and 1 (for exact agreement). The results reported in this paper are the mean and SD of the five trained models on either their corresponding test set (if drawn from the same dataset) or the entire other datasets (if dataset metadata differ) such that our model evaluations measure both in-distribution and out-of-distribution generalization behavior.

Results

Preprocessing

In order to identify the effect of training dataset on network performance, models need to first generalize well on images with nominally similar microscope conditions and sample parameters, whether the images are taken from the same dataset or are taken from two separate, but otherwise equivalent, microscope sessions. We find that choices in data preprocessing highly

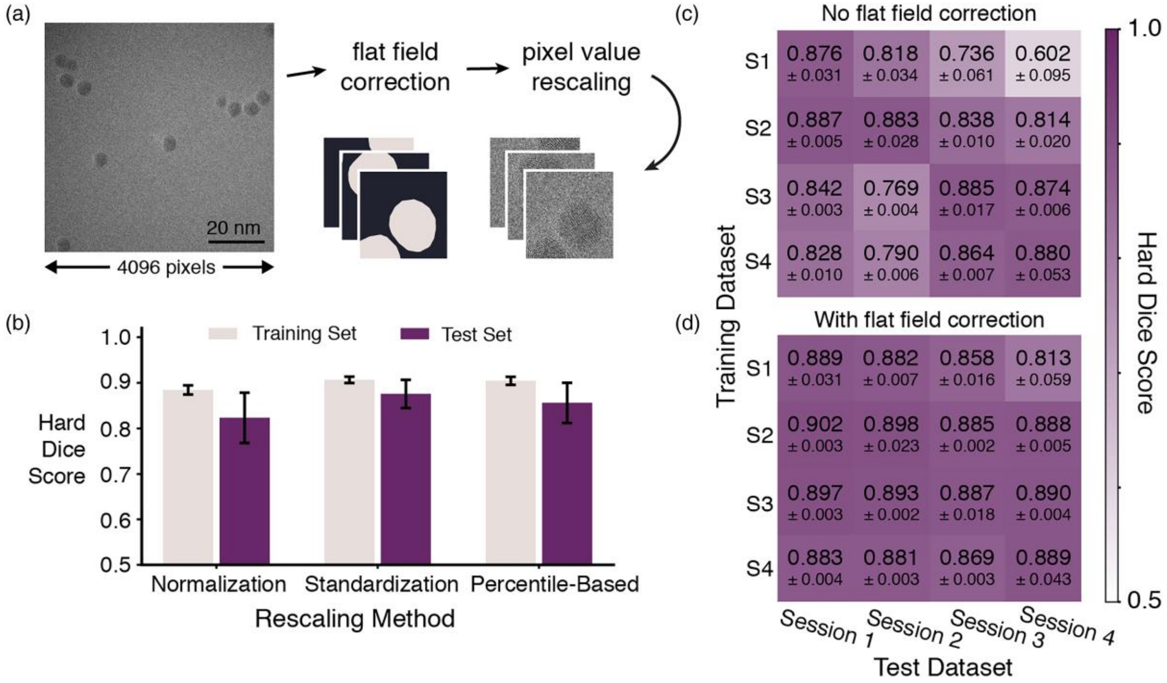

Fig. 2. Effect of data preprocessing on network generalizability. (**a**) Overview of the data preprocessing workflow, from camera output of the wide-view image to dataset creation. (**b**) The effect of pixel-value rescaling procedures (normalization, standardization, and percentile-based) on the average performance of the training set and the test set for networks trained on the same 5 nm Au nanoparticle dataset. Error bars refer to SD over five networks. (**c,d**) Confusion matrices of network performance when trained and tested on images of 5 nm Au nanoparticles taken at 0.02 nm/pixel scale and 423 e/A ^{2} dosage from four different sessions (**c**) without any flat-field correction and (**d**) with flat-field correction. Error refers to SD over five networks.

affect whether this statement holds true. Preprocessing encompasses the conversion process from raw camera data in the form of camera counts into a data format that is conducive for neural network training. There are two types of preprocessing steps: one is related to how datasets are created from the acquired data such that we can train neural networks in a memory-efficient and class-balanced manner (i.e., dividing large images into smaller patches that we feed into the neural network during training); the other is related to how image data values are converted into a standard format that enables generalization across quantitatively different camera outputs (i.e., if recorded counts are slightly different from changes in the camera gain). In this section, we will focus on the latter type of data preprocessing, and its implications for network generalization, with the key steps highlighted in Figure 2a.

Pixel-value rescaling is necessary to convert output camera data into a standard format that is robust against exact electron counts. TEM image data are outputted as an array of counts, often from a high dynamic range sensor, whose exact pixel values correspond to detector and microscope parameters like gain and exact electron dosage. We test three different rescaling methods: normalization (set image minimum to 0 and maximum to 1), standardization (set image mean to 0 and SD to 1), and a percentile-based scaling method (normalize, but ignore the pixels outside the 0.1 and 99.9 percentiles) similar to Digital Micrograph, a commonly used micrograph viewing software. Given a single set of images taken during the same session (eight images of 5 nm Au nanoparticles, which results in 211 patches), we create three datasets that have the same content, but only differ by how the data are rescaled.

We find that the choice of pixel-value rescaling method affects the generalization gap, or how well networks generalize to new images from the same microscope session. The rescaling method does not seem to noticeably affect the network's ability to converge to a solution, as evidenced by the high dice scores and low SD of the training set performance for all three rescaling methods, but does affect generalization performance to the test set (Fig. 2b). Normalization is the least robust rescaling method, having both the largest drop and variation in average test set performance relative to training set performance. Both of these trends suggest that by normalizing images, network performance is influenced by the sampling of the test set.

We attribute the performance differences across pixel rescaling methods to the larger variations in image values when normalizing, compared to more consistent nanoparticle contrast and background values when standardizing or undergoing percentile-based rescaling. TEM images often do not use the full dynamic range of the scientific sensor, and so normalization is sensitive to fluctuations in the long tail of pixel-value counts. For small datasets, these variations in image contrast across images can lead to large differences between the training and validation/test sets, while standardization and percentile-based rescaling result in more consistent pixel-value distributions between images (Supplementary Fig. S1). Due to standardization's highest performance and lowest variance, we standardize HRTEM image data for the subsequent datasets used in this paper. We do note that standardization partially relies on the assumption that the image values are normally distributed. This assumption holds mostly true for a wide-view image where the majority of the image area is amorphous substrate (see Fig. 2a), but can potentially fail for an image whose majority area is crystalline material with a bimodal pixel-value distribution from strongly diffracting lattice fringes.

In addition to consistent performance across a single microscope session or dataset, a robust algorithm should also be consistent across datasets that are nominally similar. We test our

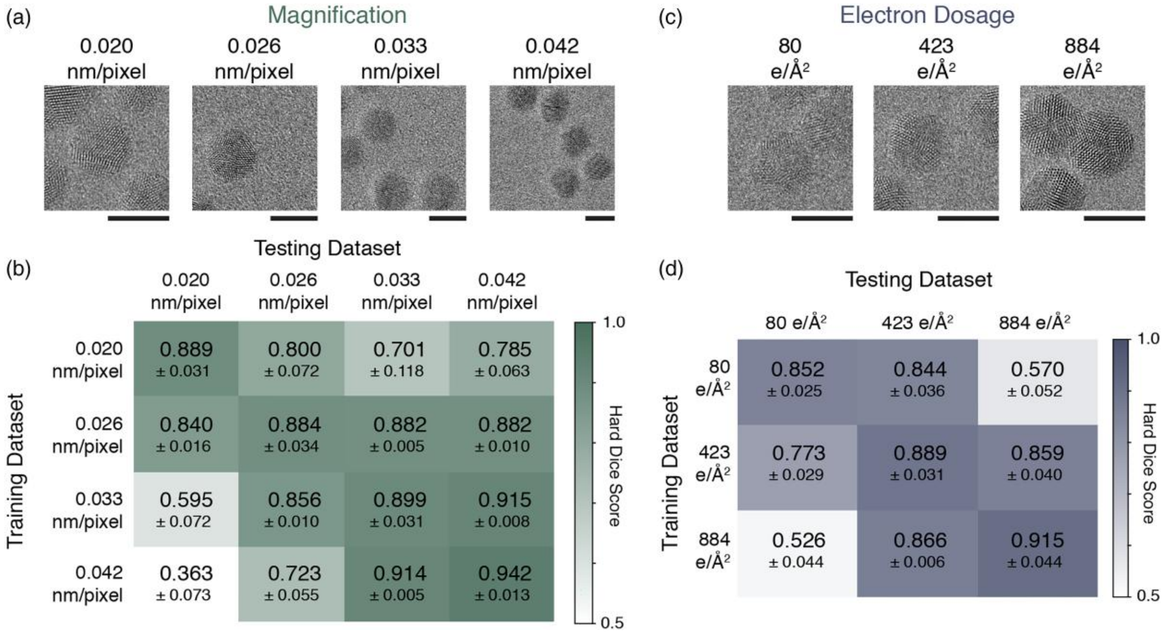

Fig. 3. Network generalizability over microscope conditions. (**a**) Sample images from the four datasets of 5 nm Au nanoparticles taken at different microscope magnifications at approximately similar dosages (420–450 e/Å ^{2}). (**b**) Confusion matrix of network performance when trained and tested with datasets taken at different magnifications. (**c**) Sample images from the three datasets of 5 nm Au nanoparticles taken at various dosage conditions but same magnification (0.02 nm/pixel). (**d**) Confusion matrix of network performance when trained and tested with datasets taken at different electron dosages. All scale bars are 5 nm.

neural networks' ability to generalize to datasets taken during four different microscope sessions but with nominally similar sample and imaging conditions (5 nm Au nanoparticles taken at 0.02 nm pixel scale with 423 e/Å ^{2} dosage). Figure 2c shows a confusion-matrix-style visualization of the networks' performance, with the diagonal elements highlighting the performance on test data taken from the same dataset, and the off-diagonal elements showcasing the performance on data from sessions different from the training dataset. These networks primarily perform well on test sets drawn from the same dataset (i.e., same microscope session) they were trained on, but fail to generalize to nominally similar data, suggesting that there is session-dependent information that the models are capturing.

By applying flat-field correction to our images, we are able to obtain better generalization performance across microscope sessions. This preprocessing step corrects for uneven illumination across the image caused by either shifts in the monochromator or incorrect gain references. As seen in Figure 2d, once the images are flat field corrected, networks generalize much better to other sessions of nominally similar data. Flat-field correction is particularly influential in our datasets because pixel-value rescaling is done per image; the correction ensures that there is less variation across patches such that patch statistics better match larger scale image statistics. Note that flat-field correction does not seem to impact how well the networks analyze the data—the diagonal elements of the confusion matrix retain similar performance regardless of flat-field correction. Therefore, flat-field correction on our datasets primarily removes session-dependent experimental artifacts (quantified in Supplementary Table S2) that affect generalization.

Generalizability Across Microscope Parameters

Microscope parameters heavily affect how an HRTEM image is formed and the subsequent observed image features. As a sanity check, we first investigate the generalizability of networks across microscope magnifications. Our networks are not expected to generalize well since lattice fringes, a key nanoparticle image feature, have a characteristic length scale and our CNNs are, by construction, not scale-invariant. We create four datasets, each of 5 nm Au nanoparticles taken at similar dosages (420–450 e/Å ^{2}) but different magnifications (Fig. 3a), and then train and test networks across the four datasets. We plot our test results in a confusion-matrix-style visualization, with the diagonal elements showing "in-distribution" behavior, and the off-diagonal elements showing "out-of-distribution" behavior; each row consists of a different set of neural networks trained on distinct experimental datasets, indicated by the row labels. As expected, neural network performance is worse on images taken at a different magnification than the training dataset, with a larger drop in performance on test sets with a greater difference in pixel scale (Fig. 3b).

From the confusion matrix, we see that generalization behavior is not necessarily symmetrical. For instance, networks trained on images taken at 0.042 nm pixel scale perform extremely poorly on images taken at 0.02 nm pixel scale, but this difference in performance is smaller vice versa. We hypothesize that this asymmetry is from the neural network using additional information beyond just the spatial frequency of lattice fringes to make its decisions (Sytwu et al., 2022) as these datasets are not just rescaled versions of each other. In HRTEM images of nanoparticles on amorphous background, nanoparticles have greater image contrast in images taken at lower magnifications (i.e., 0.042 nm/pixel) than in images taken at higher magnification (0.02 nm/pixel) which we hypothesize are due to pixel-value rescaling methods that depend on the SD of the background (i.e., standardization, percentile-based). The SD of the amorphous background is empirically smaller in low-magnification images compared to higher magnification images, which in turn amplifies the nanoparticle contrast after rescaling the pixel values of

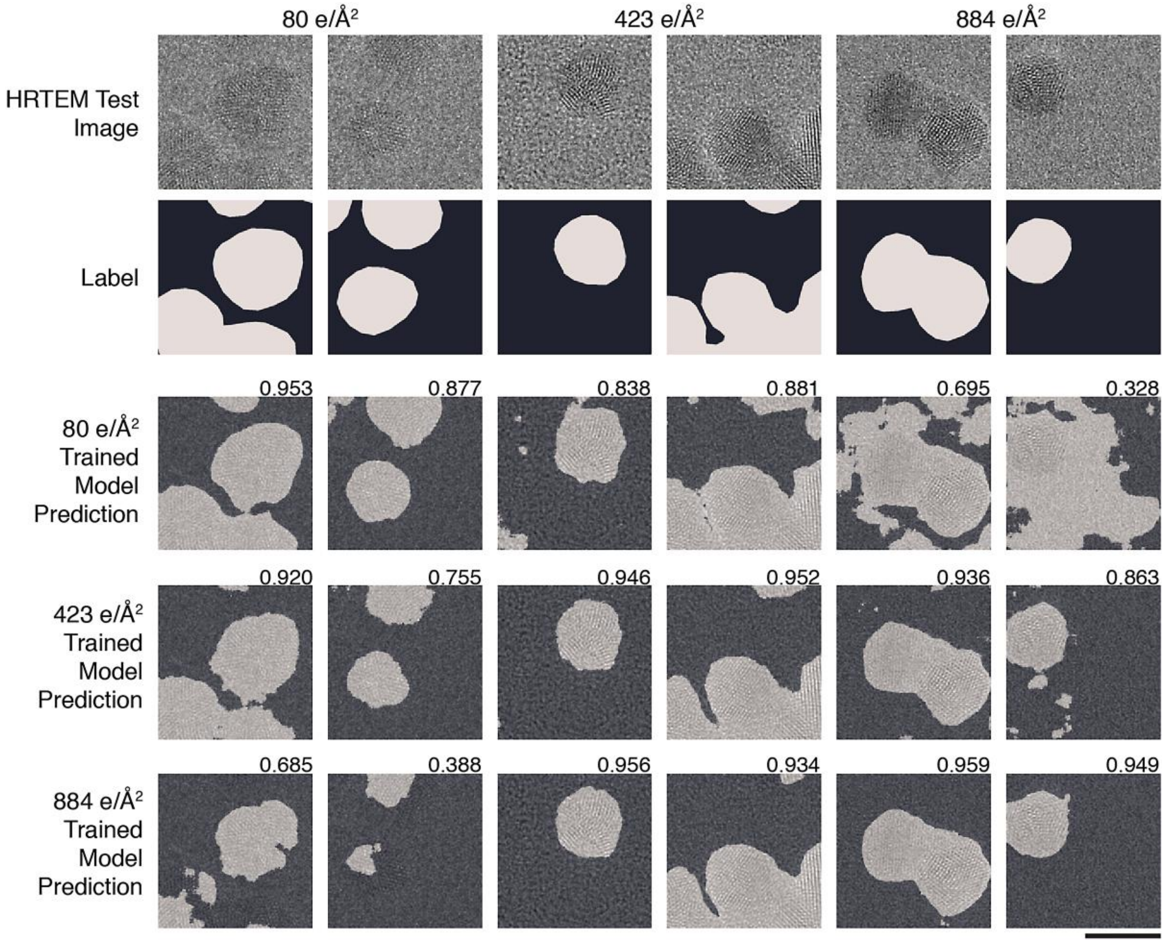

Fig. 4. Example of segmentation results from three electron dosage models. Two examples are shown from each test set (80, 423, and 884 e/Å^2). The hard dice score for each prediction is displayed at the top right corner. Scale bar is 5 nm.

the image (see Supplementary Table S3 and Supplementary Material for more details). Thus, nanoparticles are qualitatively easier to detect in low-magnification images than higher ones. The ease of distinguishing between nanoparticle and background in the lower magnification images is also noted by the overall higher performance on the 0.042 nm/pixel dataset.

In addition to magnification, we find that networks do not generalize well to datasets taken at different electron dosages, which affects the signal-to-noise ratio in the image. We again create three datasets, each of 5 nm Au nanoparticles taken at 0.02 nm pixel scale, but at three different dosages to represent a low-dose dataset (80 e/Å^2), a medium-dose dataset (423 e/Å^2), and a high-dose dataset (884 e/Å^2) (Fig. 3c). Here, we see a slightly more symmetrical confusion matrix, with all networks dropping in performance when tested on data taken at a dosage different from the training dataset (Fig. 3d). This sensitivity to the noise levels of the training dataset has also been previously observed for HRTEM nanoparticle image segmentation (Larsen et al., 2023), and can also be qualitatively analyzed by examining segmentation results from models trained on a specific dosage (Fig. 4). We observe more incorrect prediction areas when a model is tested on an image taken at a different dosage than the training dataset, with a slight tendency for networks to oversegment when tested on an image taken at a higher dosage (relative to the training set) and undersegment when tested on images taken at a lower dosage. These qualitative observations are quantitatively supported by the higher false positive rates (percentage of misclassifications where a "background" pixel is labeled as "nanoparticle") when lower dosage models are tested on higher dosage images (Supplementary Fig. S5). This suggests that evaluating on a dataset with a dosage different from the training dataset could incorrectly bias subsequent nanoparticle size analysis, but more detailed studies with a wider range of dosage values are needed to quantify these potential errors.

Generalization Across Sample Parameters

Understanding the reliability of a trained neural network across sample parameters is especially crucial for *in situ* studies and automated microscopy where microscope parameters are usually fixed but sample parameters may change or be unknown in the future. For nanoparticle datasets, one commonly varying sample parameter is nanoparticle size. We again create three datasets, each of Au nanoparticles taken at similar microscope conditions (0.02 nm/pixel at 423–425 e/Å^2) but varying in average nanoparticle diameter from 2.2 to 10 nm (Fig. 5a). While the observed lattice fringes have the same characteristic length scale in all three datasets, larger nanoparticles are thicker and therefore have greater nanoparticle contrast (both amplitude and phase-contrast) against the substrate background. When evaluating network generalization (Fig. 5b), we find that some models and datasets generalize well. All models perform equally well on the 5 nm and 10 nm datasets, but there is some variation in performance on the 2.2 nm dataset depending on the training

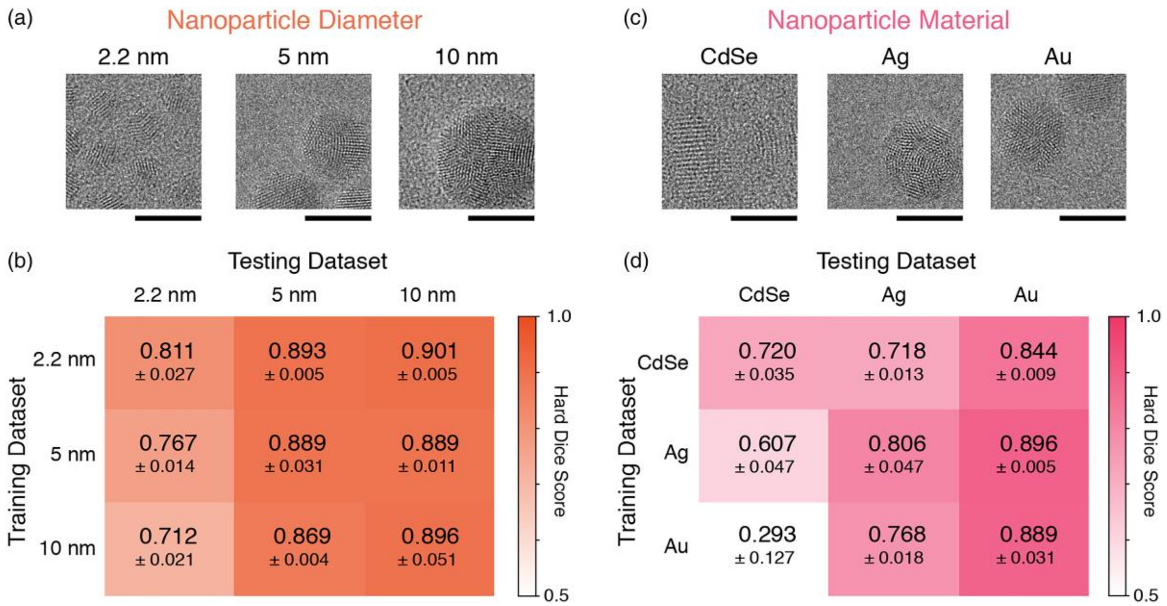

Fig. 5. Network generalizability over nanoparticle sample parameters. (**a**) Sample images from the three datasets of Au nanoparticles of either 2.2, 5, or 10 nm in diameter taken with similar microscope conditions (0.02 nm/pixel at 423–425 e/Å ^{2}). (**b**) Confusion matrix of network performance when trained and tested with datasets of Au nanoparticles with different diameters. (**c**) Sample images from the three datasets of approximately 5 nm nanoparticles of either CdSe, Ag, or Au taken with similar microscope conditions (0.02 nm/pixel at 421–423 e/Å ^{2}). (**d**) Confusion matrix of network performance when trained and tested with datasets of nanoparticles of different materials. All scale bars are 5 nm.

dataset. The models trained on 5 and 10 nm Au nanoparticle data perform worse on the 2.2 nm dataset than the models trained on 2.2 nm nanoparticles, possibly due to overdependence on amplitude contrast features which would not be significantly present for smaller nanoparticles. When normalizing for dataset difficulty, it is visually clearer that models trained on 2.2 nm data generalize better than models trained on larger nanoparticles (Supplementary Fig. S3).

Qualitative analysis also suggests that models trained on 2.2 nm data generalize better. In Figure 6, we show segmentation results from three neural networks trained on different nanoparticle diameter datasets. Here, we see much more consistent performance across all models, especially when compared to the example performance seen in Figure 4, but most notably, the 2.2 nm trained model does as well if not better than the other models at identifying the majority of nanoparticle areas. Further examination of the predicted labels of the 2.2 nm dataset suggest that the lower dice scores on the 2.2 nm dataset may be from the network identifying particles that were missed by the human labeler. In the first column of Figure 6, all three models predict that there are two nanoparticles at the bottom of the image which, upon reexamination, were missed during human labeling. The prediction from the 2.2 nm trained model, though, has a lower hard dice score (0.760) than the prediction from the 10 nm trained model (0.773), despite the cleaner and qualitatively better prediction from the 2.2 nm trained model. In total, these promising generalization results on nanoparticle size suggest that networks could be trained to perform well on image data streams without needing to know the exact nanoparticle size beforehand.

In addition to nanoparticle size, nanoparticles can also vary in their material, which leads to differences in contrast (from atomic number, Z), and nanoparticle lattice features (from lattice spacing and crystal structure). We create three datasets of approximately 5 nm nanoparticles taken at similar microscope conditions (0.02 nm/pixel at 421–423 e/Å ^{2}), but varying in material: Au, Ag, and CdSe. Au and Ag are both fcc metals with similar lattice spacings, but differ in contrast ($Z_{Au} = 79, Z_{Ag} = 47$). CdSe nanoparticles, on the other hand, can take on either a wurtzite (hexagonal) or zinc blende (fcc) structure (both appear in our sample) with average lattice spacings greater than Au and Ag, but with atomic contrast similar to Ag ($Z_{Cd} = 48$). Again, we see an imbalance in network performance depending on the dataset, with both Au- and Ag-trained networks performing well on the Au datasest, and a strong dependence on training data for the CdSe and Ag datasets (Fig. 5d). For the CdSe and Ag datasets, training on similar data does not even provide very high performance. Most interestingly, the CdSe-trained model performs decently well on the Au dataset, despite the CdSe nanoparticle regions having both different contrast and frequency information from the Au nanoparticle regions, whereas the Ag and Au-trained models perform much worse on the CdSe data. The CdSe dataset may also have worse nanoparticle contrast than expected due to the more elongated nanoparticle shape (making thickness more unknown) and beam-induced surface reconstruction which would lead to a more difficult to interpret image texture and weaker signal-to-noise.

Discussion

Overall, we find that there is potential for networks to generalize under certain sample parameters (nanoparticle size and material) but not over different microscope parameters (magnification and dosage). This suggests that pretrained neural networks could be used for data streams with controlled imaging parameters, for instance with *in situ* datasets and automated microscopy, but need to be carefully evaluated when applied to datasets of variable microscope conditions. We also find that networks trained on more difficult-to-interpret data tend to generalize to new data better than networks trained on easier-to-interpret data. In our confusion matrices, the datasets have been qualitatively ordered from lowest to highest in terms of how easily the nanoparticles

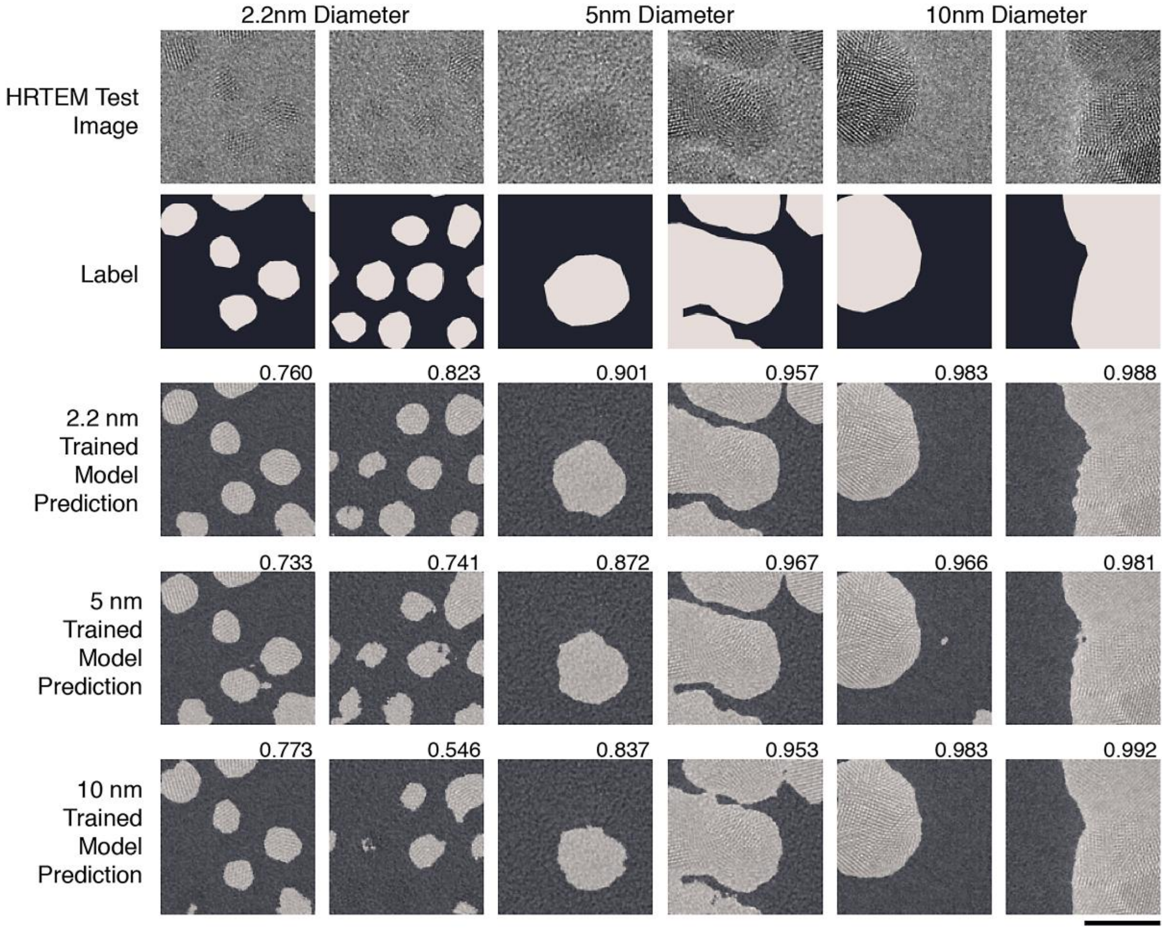

Fig. 6. Example of segmentation results from three nanoparticle diameter models. Two examples are shown from each test set (2.2, 5, and 10 nm). The hard dice score for each prediction is displayed at the top right corner. Scale bar is 5 nm.

are distinguishable, with higher nanoparticle contrast and observable lattice fringes making an image easier to interpret. Consistently, the generalization performance is worse in the lower left corner of our confusion matrices (train on easy images, test on harder images) compared to the upper right corner (train on hard images, test on easier images). Since labeling difficult-to-interpret data is prone to larger human bias and error, these results highlight the need for simulation-based or multimodal datasets with accurate ground truth information to create useful training data (Madsen et al., 2018; Vincent et al., 2021).

In the absence of collecting more data to improve the generalizability of our networks, we can alternatively mimic lower contrast and more difficult to interpret datasets by adding noise and corrupting information in the higher contrast datasets for which we have higher confidence in the labeling. Upon adding Gaussian noise to the images during training, we lower the nanoparticle contrast, but retain the lattice fringe features that denote nanoparticle regions (Fig. 7a). Note that additive Gaussian noise augmentation is a known regularization protocol to prevent overfitting (Bishop, 1995) and synthetically promote robustness (Gilmer et al., 2019).

As an example, we explore how additive noise augmentation affects generalizability across electron dosage. We train a series of models such that their training dataset of high dosage images (884 e/Å ^{2}) is augmented with additive Gaussian noise with an SD of ρ. We then evaluate the performance of these noise-augmented models on the original 884 e/Å ^{2} test set (high dose), 423 e/Å ^{2} dataset (medium dose), and 80 e/Å ^{2} dataset (low dose). As seen in Figure 7b, performance on all three datasets improve upon additive Gaussian noise augmentation, though the ideal amount of additive noise ρ depends on dataset. As expected, more additive noise is needed to improve performance on lower dosage datasets. Additionally, for all datasets, additive Gaussian noise augmentation helps networks meet or exceed the average performance of neural networks trained on experimentally collected similar data. This is surprising given that the measured noise from the OneView camera follows a scaled Poissonian distribution and is only well approximated by a Gaussian at high counts (low noise). It is unclear whether the high performance from this augmentation is from matching dataset characteristics or from regularizing decision boundaries. The optimal augmented noise level does not match the experimentally collected dataset in either nanoparticle contrast (by matching histogram medians) or noise statistics (by matching image roughness) (Supplementary Fig. S7). However, when repeating this noise augmentation procedure on the medium-dose dataset, the noise-augmented models generalize poorly to higher dose data and require less additive noise to generalize well to lower dose data, suggesting that there is some dependence on dataset characteristics (Supplementary Fig. S8). All networks degrade in performance when $\rho > 1$ SD, likely because this large noise augmentation destroys information in the image itself.

As the necessary additive Gaussian noise scale may not be known *a priori*, we alternatively set the noise augmentation

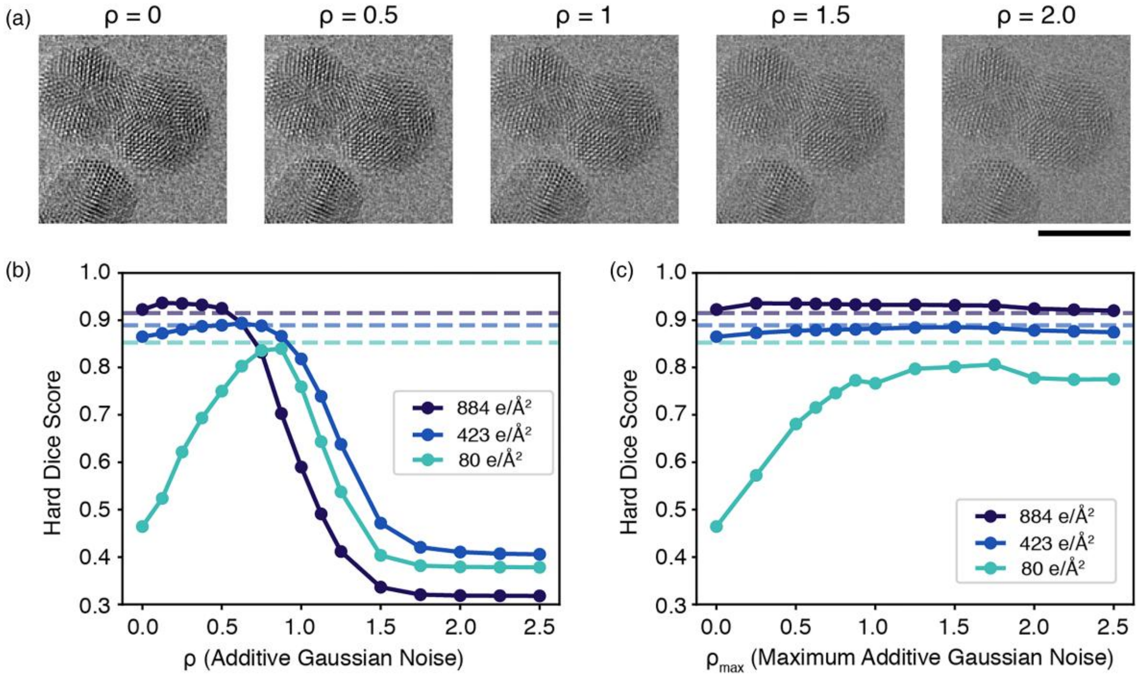

Fig. 7. The effect of additive Gaussian noise on experimental data. (**a**) Sample image from the 884 e/Å ^{2} dataset (same as in Fig. 3c) with various amounts of additive Gaussian noise of scale ρ. Scale bar is 5 nm. (**b,c**) Performance of neural networks trained on the 884 e/Å ^{2} dataset augmented with (**b**) additive Gaussian noise of scale ρ or (**c**) additive Gaussian noise with scale sampled between [0, $\rho_{\max}$] when tested on the 884 e/Å ^{2} test set, 423 e/Å ^{2} dataset, and 80 e/Å ^{2} dataset. Dotted lines indicate the average performance of the respective dataset when trained on images from the same dataset.

such that ρ is uniformly sampled between [0, $\rho_{\max}$] during training. Under this augmentation protocol, all noise-augmented-trained models perform well on the high-dose and medium-dose datasets, but none of them perform well enough on the low dose dataset to compare with low-dose trained models (Fig. 7c). These results suggest that synthetic noise augmentation could be a viable strategy for developing robust networks on HRTEM images with decent signal-to-noise, but does not work effectively to generalize to low dosage images with low signal-to-noise. Recent work has highlighted the need for more accurate noise modeling, especially for capturing the modulation transfer function and noise statistics for low dosage images in neural network training (Larsen et al., 2023), and our results similarly highlight the difficulty of generalizing to low-dosage images.

We emphasize that the focus of our results is in the data-driven generalization trends rather than absolute neural network performance, which can be affected by label error and choice of model hyperparameters. As the models in this paper are all trained from hand-labeled experimentally collected data, there is inherent human bias and error in the labels, primarily at nanoparticle edges and with lower contrast nanoparticles (as seen in Fig. 6), which affects the absolute value of the dice scores. Similarly, while our training curves suggest that our networks have converged to a local minima that enables decent performance, there is still room for improvement by fine-tuning both model and optimization hyperparameters. We argue, however, that the generalization trends that we observe are data-dependent and seem to be robust even after hyperparameter tuning; in Supplementary Fig. S6, we show the generalization performance over nanoparticle size after hyperparameter tuning model parameters, and while the overall dice scores are slightly different, the generalization trends are the same as Figure 5b. While the overall generalization performance of a neural network can also be affected by even greater changes in architecture and optimization, empirically these generalization trends have been found to be largely dictated by the relationships between data (Miller et al., 2021), similar to what we have seen in this study.

Finally, the observed sensitivity to data preprocessing suggests that we need a closer examination as to how we convert raw scientific data into datasets for machine-learning and other data-driven methods. While compressed digital images are easier to share, there needs to be greater transparency on how color mapping was performed, which affects image contrast values, visibility of outliers, and potentially leads to dataset biases (Zhong et al., 2021). Even when there is no loss in data quality from compression, there can be unintended consequences from choices made during pixel-value rescaling, specifically related to the nanoparticle contrast, as seen with the variety of in-distribution generalization gaps seen in Figure 2b and with our magnification dataset in Figure 3a. Given this sensitivity to preprocessing, care needs to be taken to not just optimize but also create reproducible image recording and preprocessing procedures (Aaron & Chew, 2021). To this end, we have not only made our processed datasets for all of our models publicly available but also the raw camera data such that preprocessing steps can be explored (see the Data Availability section and Sytwu et al., 2023). By sharing the raw camera data rather than digital images, we hope to invigorate research into the necessary data preprocessing steps for robust algorithms that work on data from any experiment.

Conclusions

We investigated how training dataset creation affects neural network segmentation performance on HRTEM images of nanoparticles. We find that choices in data preprocessing, or the conversion from raw camera data to a machine-learning-ready dataset, heavily impacts the ability for networks to generalize to new data. Overall, we find that our trained neural networks are not generalizable across microscope parameters like magnification and electron dosage, which correspond with changing image features like feature size and signal-to-noise ratio. However, networks are more generalizable across sample parameters like nanoparticle diameter and certain nanoparticle materials, which corresponds with image features like nanoparticle contrast and lattice fringe frequency. These results give insight into the experimental conditions under which we can expect trained neural networks to be reliable, and suggest the varieties of data needed for generalizable neural networks.

Availability of Data and Materials

All processed datasets, raw image data, and corresponding labels used in this paper are available in the Dryad Digital Repository, at https://doi.org/10.7941/D1SP93 (Sytwu et al., 2023). The raw image data are also available via Foundry-ML at https://doi.org/10.18126/z4mr-xwk5. Code and Jupyter notebooks on dataset creation and model training/testing, trained model weights, and more visualizations of our results are available at https://github.com/ScottLabUCB/HRTEM-Generalization.

Supplementary Material

To view supplementary material for this article, please visit http://academic.oup.com/mam/article-lookup/doi/10.1093/mam/10.1093/micmic/ozae001supplementary-data.

Acknowledgments

TEM imaging was done at the Molecular Foundry, which is supported by the Office of Science, Office of Basic Energy Sciences, of the US Department of Energy under Contract No. DE-AC02-05CH11231. This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

Financial Support

K.S. was supported by an appointment to the Intelligence Community Postdoctoral Research Fellowship Program at Lawrence Berkeley National Laboratory administered by Oak Ridge Institute for Science and Education (ORISE) through an interagency agreement between the US Department of Energy and the Office of the Director of National Intelligence (ODNI). L.R.D. was supported by the US Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Department of Energy Computational Science Graduate Fellowship under Award Number DE-SC0021110. This work was also funded by the US Department of Energy in the program "4D Camera Distillery: From Massive Electron Microscopy Scattering Data to Useful Information with AI/ML".

Conflict of Interest

The authors declare that they have no competing interest.

References

Aaron J & Chew T-L (2021). A guide to accurate reporting in digital image processing–can anyone reproduce your quantitative analysis? *J Cell Sci* **134**(6), jcs254151. https://doi.org/10.1242/jcs.254151

Bishop CM (1995). Training with noise is equivalent to Tikhonov regularization. *Neural Comput* 7(1), 108–116. https://doi.org/10.1162/neco.1995.7.1.108

Gilmer J, Ford N, Carlini N & Cubuk E (2019). Adversarial examples are a natural consequence of test error in noise. In *International Conference on Machine Learning*, pp. 2280–2289. PMLR.

Groschner CK, Choi C & Scott MC (2021). Machine learning pipeline for segmentation and defect identification from high-resolution transmission electron microscopy data. *Microsc Microanal* **27**(3), 549–556. https://doi.org/10.1017/S1431927621000386

He K, Zhang X, Ren S & Sun J (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778. New York: IEEE.

Kaufmann K & Vecchio KS (2021). An acquisition parameter study for machine-learning-enabled electron backscatter diffraction. *Microsc Microanal* **27**(4), 776–793. https://doi.org/10.1017/S1431927621000556

Labelbox. Online, 2023. Available at https://labelbox.com (retrieved September 20, 2021).

Larsen MHL, Lomholdt WB, Valencia CN, Hansen TW & Schiøtz J (2023). Quantifying noise limitations of neural network segmentations in high-resolution transmission electron microscopy. *Ultramicroscopy*. **253**, 113803. https://doi.org/10.1016/j.ultramic.2023.113803

Li K, DeCost B, Choudhary K, Greenwood M & Hattrick-Simpers J (2023). A critical examination of robustness and generalizability of machine learning prediction of materials properties. *NPJ Comput Mater* **9**(1), 55. https://doi.org/10.1038/s41524-023-01012-9

Liu Z, Lian T, Farrell J & Wandell BA (2020). Neural network generalization: The impact of camera parameters. *IEEE Access* **8**, 10443–10454. https://doi.org/10.1109/Access.6287639

Madsen J, Liu P, Kling J, Wagner JB, Hansen TW, Winther O & Schiøtz J (2018). A deep learning approach to identify local structures in atomic-resolution transmission electron microscopy images. *Adv Theory Simul* **1**(8), 1800037. https://doi.org/10.1002/adts.201800037

Miller JP, Taori R, Raghunathan A, Sagawa S, Koh PW, Shankar V, Liang P, Carmon Y & Schmidt L (2021). Accuracy on the line: On the strong correlation between out-of-distribution and in-distribution generalization. In *International Conference on Machine Learning*, pp. 7721–7735. PMLR.

Mohan S, Manzorro R, Vincent JL, Tang B, Sheth DY, Simoncelli EP, Matteson DS, Crozier PA & Fernandez-Granda C (2022). Deep denoising for scientific discovery: A case study in electron microscopy. *IEEE Trans Comput Imaging* **8**, 585–597. https://doi.org/10.1109/TCI.2022.3176536

Recht B, Roelofs R, Schmidt L & Shankar V (2019). Do ImageNet classifiers generalize to ImageNet? In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, Chaudhuri K & Salakhutdinov R (Eds.), pp. 5389–5400. PMLR.

Ronneberger O, Fischer P & Brox T (2015). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241. Cham: Springer.

Sadre R, Ophus C, Butko A & Weber GH (2021). Deep learning segmentation of complex features in atomic-resolution phase-contrast transmission electron microscopy images. *Microsc Microanal* **27**(4), 804–814. https://doi.org/10.1017/S1431927621000167

Shen Z, Liu J, He Y, Zhang X, Xu R, Yu H & Cui P (2021). Towards out-of-distribution generalization: A survey. arXiv. arXiv:2108.13624, preprint: peer reviewed.

Sytwu K, Groschner C & Scott MC (2022). Understanding the influence of receptive field and network complexity in neural network-guided TEM image analysis. *Microsc Microanal* **28**(6), 1896–1904. https://doi.org/10.1017/S1431927622012466

Sytwu K, Rangel DaCosta L & Scott MC (2023). Segmented high-resolution transmission electron microscopy images of nanoparticles. Dryad. Available at https://doi.org/10.7941/D1SP93 (retrieved July 31, 2023).

Torralba A & Efros AA (2011). Unbiased look at dataset bias. In *CVPR 2011*, pp. 1521–1528. New York: IEEE.

Vincent JL, Manzorro R, Mohan S, Tang B, Sheth DY, Simoncelli EP, Matteson DS, Fernandez-Granda C & Crozier PA (2021). Developing and evaluating deep neural network-based denoising for nanoparticle TEM images with ultra-low signal-to-noise. *Microsc Microanal* **27**(6), 1431–1447. https://doi.org/10.1017/S1431927621012678

Wei J, Blaiszik B, Scourtas A, Morgan D & Voyles PM (2023). Benchmark tests of atom segmentation deep learning models with a consistent dataset. *Microsc Microanal* **29**(2), 552–562. https://doi.org/10.1093/micmic/ozac043

Yao L, Ou Z, Luo B, Xu C & Chen Q (2020). Machine learning to reveal nanoparticle dynamics from liquid-phase TEM videos. *ACS Cent Sci* **6**(8), 1421–1430. https://doi.org/10.1021/acscentsci.0c00430

Zhong X, Gallagher B, Eves K, Robertson E, Mundhenk TN & Han TY-J (2021). A study of real-world micrograph data quality and machine learning model robustness. *NPJ Comput Mater* **7**(1), 161. https://doi.org/10.1038/s41524-021-00616-3

TESCAN
TESCAN
TESCAN AMBER X