npj | computational materials

Article

Published in partnership with the Shanghai Institute of Ceramics of the Chinese Academy of Sciences

https://doi.org/10.1038/s41524-025-01702-6

Addressing data scarcity in nanomaterial segmentation networks with differentiable rendering and generative modeling

Dennis Possart[1,2] ✉, Leonid Mill[3,4], Florian Vollnhals[5], Tor Hildebrand[6], Peter Suter[6], Mathis Hoffmann[3], Jonas Utz[2], Daniel Augsburger[1], Mareike Thies[3], Mingxuan Gu[3], Fabian Wagner[3], George Sarau[1,5,7], Silke Christiansen[1,5,7,8] & Katharina Breininger[2,9]

Nanomaterials' properties, influenced by size, shape, and surface characteristics, are crucial for their technological, biological, and environmental applications. Accurate quantification of these materials is essential for advancing research. Deep learning segmentation networks offer precise, automated analysis, but their effectiveness depends on representative annotated datasets, which are difficult to obtain due to the high cost and manual effort required for imaging and annotation. To address this, we present DiffRenderGAN, a generative model that produces annotated synthetic data by integrating a differentiable renderer into a Generative Adversarial Network (GAN) framework. DiffRenderGAN optimizes rendering parameters to produce realistic, annotated images from non-annotated real microscopy images, reducing manual effort and improving segmentation performance compared to existing methods. Tested on ion and electron microscopy datasets, including titanium dioxide (TiO_2), silicon dioxide (SiO_2), and silver nanowires (AgNW), DiffRenderGAN bridges the gap between synthetic and real data, advancing the quantification and understanding of complex nanomaterial systems.

Nanomaterials are ubiquitous and exhibit unique properties that are often dictated by their size, shape, and surface characteristics. These attributes influence not only their performance in technological applications but also their interactions within biological and environmental systems. A precise understanding of these parameters is therefore critical across fields, whether the goal is to optimize material properties for advanced technologies or to assess potential risks in environmental and health contexts. For example, titanium dioxide (TiO_2) and silicon dioxide (SiO_2) nanoparticles are used in a wide range of applications, from nanomedicine[1,2] to photocatalysis[3] and wastewater treatment[4]. Furthermore, silver nanowires (AgNWs) are promising candidates for indium-free transparent electrodes[5,6] and are widely studied in terms of degradation and performance[7–10].

To effectively analyze nanomaterials, automated methods are necessary, particularly when dealing with complex particle agglomerates and large numbers of particles. Deep learning segmentation networks have emerged as powerful tools in this regard, transforming quantitative analysis in microscopic imaging from traditional subjective methods to precise and automated approaches[11]. For example, these networks now offer unprecedented insight into pathological findings[12,13] and material production processes[14,15].

However, their ability to generalize to novel, unseen data critically depends on the availability of representative training datasets[16], as these datasets determine the data distribution from which diverse and class-defining features are derived[17]. If the training data distribution insufficiently

[1]Correlative Microscopy and Materials Data, Fraunhofer Institute for Ceramic Technologies and Systems, Forchheim, Germany. [2]Department Artificial Intelligence in Biomedical Engineering, Friedrich-Alexander-Universität Erlangen-Nürnberg, Erlangen, Germany. [3]Pattern Recognition Lab, Friedrich-Alexander-Universität Erlangen-Nürnberg, Erlangen, Germany. [4]MIRA Vision Microscopy GmbH, Göppingen, Germany. [5]Institute for Nanotechnology and Correlative Microscopy, Forchheim, Germany. [6]Lucid Concepts AG, Zurich, Switzerland. [7]Emeritus-Gruppe Leuchs, Max-Planck-Institut für die Physik des Lichts, Erlangen, Germany. [8]Institute of Experimental Physics, Freie Universität Berlin, Berlin, Germany. [9]Center for AI and Data Science, Julius-Maximilians-Universität Würzburg, Würzburg, Germany. ✉e-mail: dennis.possart@fau.de

represents the problem at hand, models will perform unsatisfactorily[17,18]. In microscopic imaging, several challenges hinder the acquisition of comprehensive datasets, including high equipment costs, reliance on highly specialized personnel, and the labor-intensive nature of manual image annotation.

To address these challenges, researchers have increasingly turned to data synthesis methods. Generative adversarial networks (GANs) have shown significant potential in generating synthetic annotated data in an unsupervised manner, effectively capturing the essence of real data[19–22]. For example, Rühle et al.[23] successfully utilized WassersteinGANs[24] and CycleGANs[25] to synthesize annotated Scanning Electron Microscopy (SEM) images for the identification and segmentation of TiO_2 nanoparticles. Other approaches have explored the incorporation of prior knowledge into the data synthesis process[26], such as expert-guided image rendering[27,28]. Mill et al.[28] demonstrated this technique by simulating helium-ion microscopy (HIM) images of SiO_2 and TiO_2 nanoparticles to train expressive segmentation networks.

Although synthetic data was effectively used in the studies of Rühle et al. and Mill et al.[23,28], evaluation results showed that segmentation models trained on synthetic data generally underperformed in most metrics compared to those trained on real data, indicating a domain gap in synthetic data. For the GAN-based method of Rühle et al., reduced segmentation performance could be attributed to factors such as visual artifacts, training instability, and inaccuracies in the synthetic labels. In contrast, Mill et al.'s rendering approach may have exhibited lower segmentation performance due to the omission of class-important features that exceed the identification and rendering capabilities of domain experts.

Recent advances in differentiable rendering offer new potential by enabling the automatic optimization of reality-replicating 3D models using gradient descent methods[29]. This minimizes reliance on manual expertise while enhancing the realism of synthetic images. Building on this potential and the unsupervised training capabilities of GANs, we combine both techniques and introduce DiffRenderGAN, a novel generative model that integrates a differentiable renderer within a GAN framework. Using nanoparticle 3D models, such as meshes, and a transformation matrix containing positional and scaling information to arrange these meshes realistically, DiffRenderGAN learns distributions of textural rendering parameters that simulate materials from a given real nanoparticle dataset. This parametric representation enables the generation of synthetic, annotated images that closely mirror real-measured data. These images can then be used to train segmentation networks effectively, facilitating the identification and quantification of nanoparticles in measured microscopy images.

In Fig. 1, we summarize the contributions of this work. This paper presents DiffRenderGAN and demonstrates its application across various microscopy datasets, including those from Mill et al.[28] (SiO_2, TiO_2 in HIM) and Rühle et al.[23] (TiO_2 in SEM), as well as an AgNW dataset using multibeam SEM. We evaluate DiffRenderGAN by comparing the synthetic data it generates with other methods, training segmentation models only on synthetic data, and testing them on real microscopy images. For the AgNW dataset, where ground truth annotations are unavailable, we assess DiffRenderGAN qualitatively.

Our results demonstrate that DiffRenderGAN effectively optimizes parameters for realistic image generation, reducing manual effort to selecting target meshes and training parameters. Our method meets or exceeds the performance of existing methods, achieving higher scores across key segmentation metrics. These results highlight the potential of DiffRenderGAN as a powerful tool for generating synthetic multimodal microscopy data, reducing the domain gap in synthetic images, and

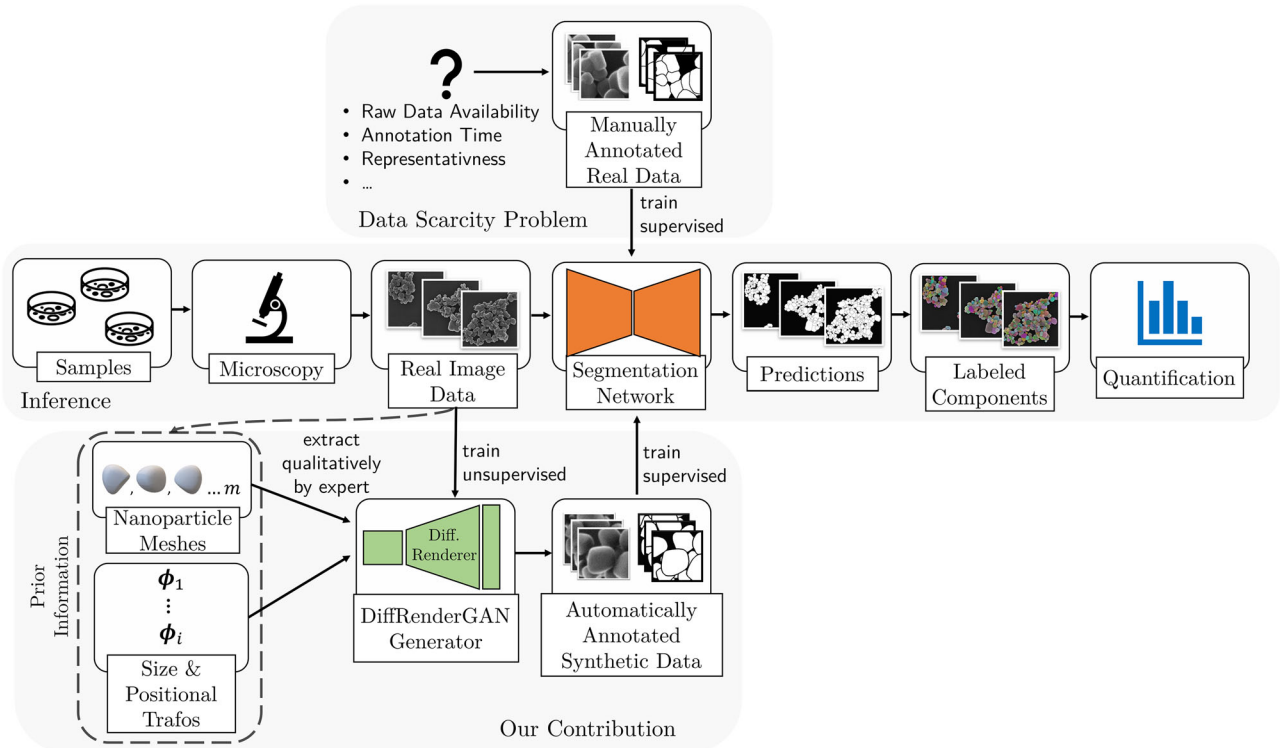

Fig. 1 | Addressing training data scarcity in deep segmentation networks for quantitative nanomaterial analysis through synthetic data generation. This figure highlights the key contributions of our work by illustrating the data bottlenecks inherent in the standard methodology used for training models in nanoparticle micrograph quantification. Our contribution aims to address three primary objectives: (1) to present an image synthesis method applicable across various microscopy modalities for the analysis of materials with diverse morphologies, (2) to minimize the need for expert intervention, and (3) to reduce or eliminate the representativeness gap between synthetic and real data, as observed in previous studies, enabling more efficient training of deep segmentation networks for improved analysis of complex nanomaterial systems. It is important to note that our goal is not to generate a physically accurate simulation of materials but rather to conduct a simulation that produces images capturing the characteristics necessary for training a generalizing segmentation network.

advancing the analysis and understanding of complex nanomaterial systems.

Results

Leveraging differentiable rendering for enhanced generative modeling

Our image synthesis method integrates the principles of image rendering and GANs. Image rendering involves transforming a virtual 3D scene into a realistic 2D digital image from a specified perspective[30]. The virtual 3D scene is defined by parameters such as meshes (e.g., 3D nanoparticle models) and textures attached to them, referred to in this work as bidirectional scattering distribution functions (BSDFs), which simulate material properties like diffuse or dielectric behavior. In addition, light sources are included to define observable emissions. Formally, we denote the rendering process as f_r, which generates an image I_r from the virtual scene expressed by Θ:

$$I_r = f_r(\Theta). \tag{1}$$

The interested reader is referred to Kajiya et al.[31] for a detailed definition and description of the rendering function f_r. Rendering has been used in computer vision to create synthetic datasets for training machine learning models by integrating expert knowledge into the design of the virtual scene[27,28,32].

One key advantage of rendering-based synthetic data is that annotation masks can be automatically extracted using unique identifiers assigned to each mesh in the virtual scene. Expert-guided image rendering prevents visual artifacts and labeling inaccuracies that are common in CycleGAN applications[33–35]. However, the expert-driven process of creating synthetic images is time-consuming, and key features might be overlooked in complex reference images. Therefore, a data-driven approach may be more desirable.

Differentiable rendering makes this possible by enabling the calculation of $\frac{\partial I_r}{\partial \Theta}$, allowing for iterative optimization of virtual scene parameters[36]. Using methods such as stochastic gradient descent or Adam[37,38], parameters can be adjusted to minimize an objective function, such as the mean-squared error between a rendered image and a target image. Replicating real data using a differentiable renderer presents significant challenges, particularly when working with large datasets of nanoparticle images. Achieving a realistic representation of each observed nanoparticle in images necessitates the accurate reconstruction and positioning of meshes, a process that becomes increasingly complex as the number of particles in the dataset grows. To address this challenge, we employ GANs, which are capable of generating realistic and diverse data distributions rather than exact replicas.

GANs, introduced by Goodfellow et al.[19], consist of two neural networks: a generator $G(z)$ that maps a random noise vector z from a distribution p_z into a synthetic image, and a discriminator $D(x)$ that classifies images as real or fake, where x denotes a real sample from the distribution p_{data}. The generator aims to produce images that are indistinguishable from real data, while the discriminator is tasked with effectively differentiating between real and synthetic images. The adversarial process is formulated as a min-max optimization problem:

$$\min_G \max_D \mathbb{E}_{x\sim p_{\text{data}}(x)}\left[\log D(x)\right] + \mathbb{E}_{z\sim p_z(z)}\left[\log(1 - D(G(z)))\right] \tag{2}$$

By combining the unsupervised training capabilities of GANs with the controllability of differentiable rendering and automatic mask extraction, we developed DiffRenderGAN, which integrates a differentiable renderer into the GAN's generator. This integration enables the generation of highly realistic synthetic images achieved without reconstruction. Simultaneously, the controlled rendering environment mitigates common visual artifacts, such as checkerboard patterns often observed in CycleGAN applications[25], thereby ensuring higher-quality and more consistent outputs.

Optimizing all virtual scene parameters Θ to visually simulate real nanoparticles, including their morphologies, is computationally demanding. To simplify this process, our generator focuses on optimizing textural parameters θ_{BSDF} that mimic the material properties observed in SEM and HIM imaging. Assumptions regarding morphologies, size distribution, and placement of reference nanomaterials are provided by experts before training to guide DiffRenderGAN. This assumption-based strategy allows for a realistic arrangement of meshes without the need for direct optimization of their shapes and positions. We define the virtual scene parameter space Θ as:

$$\Theta = \begin{pmatrix} \theta_{\text{BSDF}} \\ \theta_{\text{other}} \end{pmatrix}, \tag{3}$$

where θ_{BSDF} includes all optimized BSDF parameters, while θ_{other} encompasses non-optimizable BSDF parameters and all other scene parameters, including those related to geometry, position, and size.

Before training DiffRenderGAN (see Fig. 2), an expert-guided process is employed to model a collection of n particle meshes that reflect the shape properties of nanoparticles observed in real images (detailed in Nanoparticle Mesh Modeling). The sizes and positional arrangements of the meshes are selected from distributions such as normal, lognormal, or bimodal. Mesh placement can either be random or agglomerated, utilizing a Poisson Disk-based sampling algorithm for clustering[39]. Subsequently, based on the selected placement and scale strategy, a transformation tensor

$$\Phi = \{\phi_i | \phi_i \in \mathbb{R}^{n\times 4},\ i = 0, 1, \ldots, m-1\} \tag{4}$$

is computed, where each subtensor ϕ_i contains spatial coordinates and a scaling factor for each of the n meshes. The tensor Φ encodes m different nanoparticle constellations, defining the synthetic image sampling size with varying mesh arrangements used during training. We detail the computation of the transformation tensor in "Transformation computation".

The architecture of DiffRenderGAN's generator is organized into three modules, as depicted in Fig. 2. The first module, a fully connected network (FCN), denoted as f_{fcn}, takes a uniformly randomly selected $\phi_i \sim \mathcal{U}(\Phi)$, serving as a distinct mapping. This is analogous to the role of the randomly sampled noise vector z in vanilla GANs (as illustrated in equation (2)). The FCN regresses θ_{BSDF} and a noise scale σ, which is subsequently used in the generator's final module to introduce learnable Gaussian noise:

$$(\theta_{\text{BSDF}}, \sigma) = f_{\text{fcn}}(\phi_i). \tag{5}$$

For regularization and training stability, f_{fcn} produces parameter estimates in [0, 1]. We then rescale and clip these values for both the BSDF parameters θ_{BSDF} and the noise deviation σ so that they lie in their respective physically and render-environmental feasible ranges. The specific limits used in this work, including noise, stage, and particle mesh BSDF boundaries, are provided in Supplementary Table 1. In the second module, a virtual scene is dynamically created where the collection of n expert-generated particle meshes is positioned and scaled according to ϕ_i. The latest BSDF values from θ_{BSDF} are applied to both the nanoparticle meshes and a rectangular stage mesh located beneath them. The virtual scene is then passed to the differentiable renderer, f_r, to generate a synthetic image.

To simulate real-world imaging conditions, the third module, f_{noise}, adds zero-centered Gaussian noise scaled by σ to the rendered image. This step aims to replicate the noise present in real images, making the synthetic output more realistic. The final synthetic image I_{synth} is evaluated by the discriminator for its realism, allowing for gradient computation via back-propagation to update weights in the generator's f_{fcn}[40]. The generator's overall functionality is summarized as:

$$I_{\text{synth}} = G(\phi_i) = f_{\text{noise}}(f_r(f_{\text{fcn}}(\phi_i), \theta_{\text{other}}), \sigma). \tag{6}$$

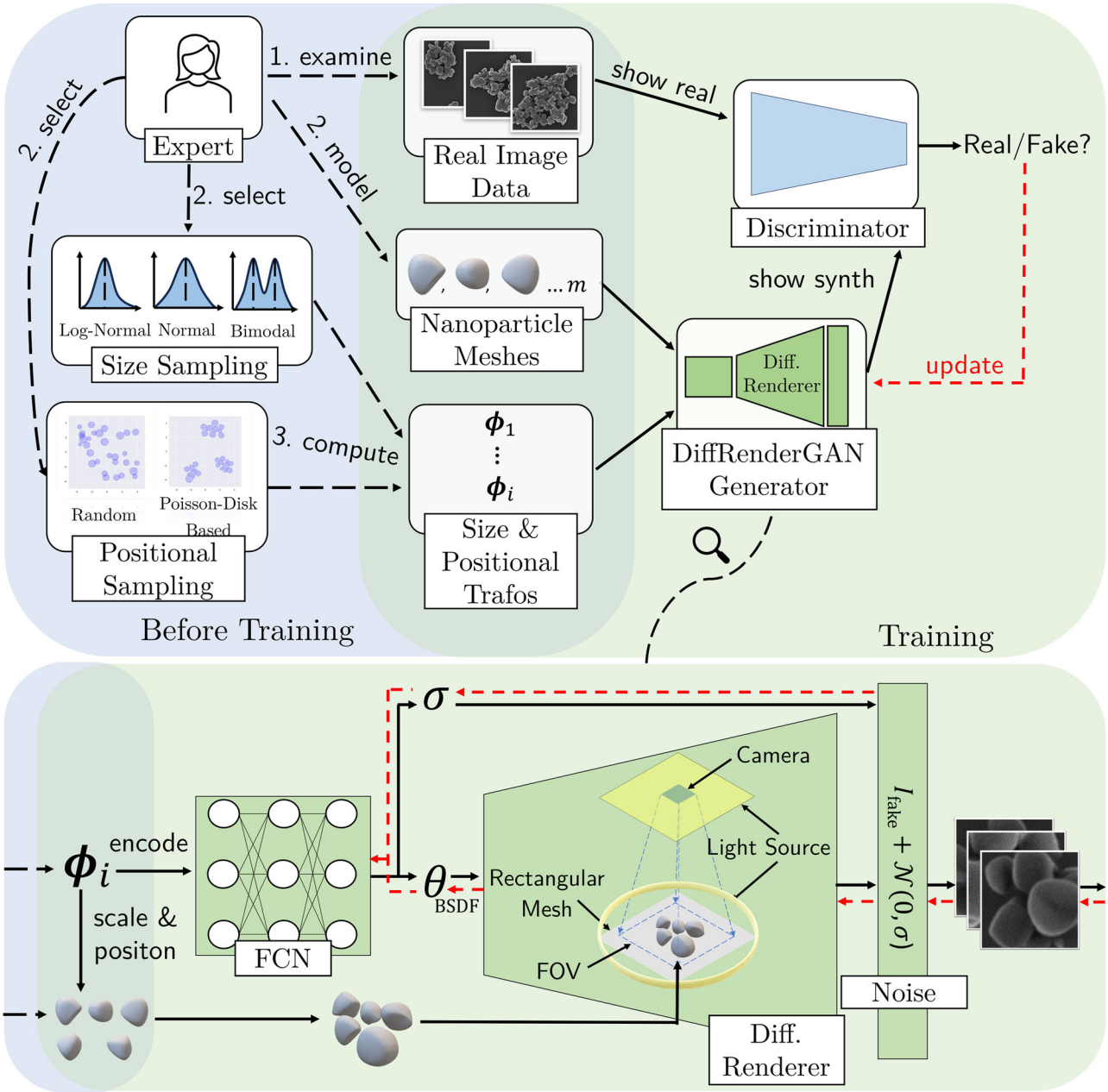

Fig. 2 | DiffRenderGAN training procedure. This figure illustrates the overall working process of our proposed methodology. We provide a detailed visualization of the novel generator architecture, highlighting its key components and how it integrates differentiable rendering to synthesize realistic, annotated microscopy images from unannotated inputs. Domain experts create target nanomaterial meshes to match the morphology of real particle systems. Scale and placement parameters are used to compute a transformation matrix for training. The meshes and transformation matrix serve as input to the DiffRenderGAN model. During image generation, a slice of the matrix is processed by a 5-layer Fully Connected Network (FCN) to predict BSDF parameters and noise scale. These parameters are passed to a differentiable renderer, which uses a virtual scene with scaled and positioned meshes to create the final synthetic nanomaterial image. A technical description of DiffRenderGAN's modules is provided in "Model design". For visualization purposes, the virtual scene used by the differentiable renderer is shown in a simplified form. The actual structure can be found in the supplementary information of this paper, with a technical summary stated in "Virtual scene design".

Within the adversarial training framework, DiffRenderGAN's generator and discriminator engage in the following adversarial process:

$$\min_G \max_D \mathbb{E}_{x\sim p_{\text{data}}(x)}[\log D(x)] + \mathbb{E}_{\phi_i\sim\mathcal{U}(\Phi)}[\log(1 - D(G(\phi_i)))]. \quad (7)$$

The available data from each image case introduced in "Introduction" is split into ~80% for training and 20% for testing (later used in "Deep learning-based segmentation of nanoparticles trained on synthetic images"). Details of the image data acquisition and a sample description are provided in "Image acquisition and processing". Each image of the training dataset is cropped into overlapping patches of size 256×256 pixels. DiffRenderGAN is then trained on image patches that contain at least three fully displayed particles while avoiding repetitive particle patches.

At the same time, we demonstrate an effective use of image patches that do not contain particles but still provide valuable background information, such as artifacts, which do not necessitate additional annotation for particle segmentation tasks. We extract 200 of these patches for each dataset, which we later use to supplement our synthetic datasets.

The training process is monitored using the Fréchet Inception Distance (FID) score, a state-of-the-art metric that measures the feature distance between the generated synthetic images and real images[41]. To determine the best epoch, we compare the five epochs with the lowest FID scores and select

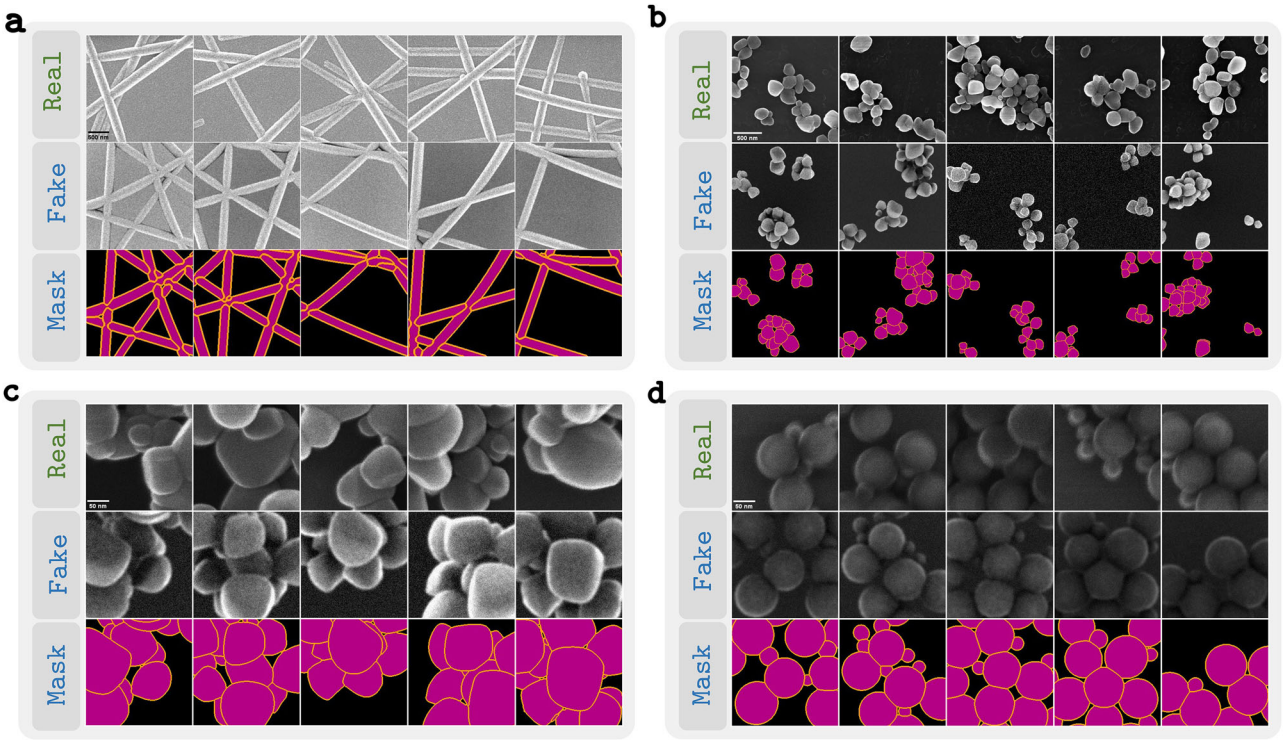

Fig. 3 | Comparison of real and synthetic image patches with corresponding segmentation masks. This figure showcases representative synthetic images produced after DiffRenderGAN optimization for each microscopy case. In each figure section, the top row shows real images used to train DiffRenderGAN, the middle row depicts synthetic images, and the bottom row shows the corresponding segmentation masks, highlighting material classes (purple) and boundaries (orange). These synthetic image-mask pairs serve as training data for multiclass segmentation networks, as demonstrated in (deep learning-based segmentation of nanoparticles trained on synthetic images). **a** AgNW: trained using 10 bent cone meshes, choosing for transformation computation random placement in 2D and a lognormal size distribution. **b** TiO_2 in SEM from Rühle et al.[23]: trained using 40 cubically deformed meshes, choosing for transformation computation Poisson Disk-based placement in 3D and a lognormal size distribution. **c** TiO_2 in HIM from Mill et al.[28]: trained using 15 cubically deformed meshes, choosing for transformation computation Poisson Disk-based placement in 3D and a lognormal size distribution. **d** SiO_2 in HIM from Mill et al.[28]: trained using 20 sphere meshes, choosing for transformation computation Poisson Disk-based placement in 3D and a lognormal size distribution.

the one that demonstrates a broader distribution of learned parameters. This ensures a balance between a low FID score and diversity in the learned parametric distributions, preventing the selection of a mode-collapsed model and ensuring that the final model produces high-quality and varied synthetic data.

In Fig. 3, we present samples of the synthetic data generated using the trained models for each material case, along with their automatically computed mask images (see "Model inference"). An overview of all experiments is provided in Supplementary Table 1. Parameter progressions during training, along with corresponding synthetic images after optimization, are presented for each material case in Supplementary Figs. 2–5.

Deep learning-based segmentation of nanoparticles trained on synthetic images

After training DiffRenderGAN on the four image cases, we assessed its effectiveness by training segmentation models on each respective synthetic dataset. For three of these cases (TiO_2 HIM, SiO_2 HIM, and TiO_2 SEM), synthetic data produced by previously published methods are available for comparison[23,28]. For the AgNW case, where no alternative synthetic data or ground truth annotations are available, we performed a qualitative assessment of our synthetic data to demonstrate its effectiveness for rod-like nanoparticles.

To comprehensively evaluate segmentation performance across different aspects, we employ three key metrics: the dice similarity coefficient (DSC), average precision (AP) at an Intersection-over-Union (IoU) threshold of 50%, and panoptic quality (PQ). The DSC measures the overlap between predicted and ground truth segmentation masks[42], providing a direct assessment of segmentation accuracy. AP quantifies the precision of object localization at a fixed IoU threshold, reflecting a model's ability to correctly detect and delineate nanoparticles[43]. Lastly, PQ integrates both segmentation and object detection accuracy into a single metric, offering an evaluation of both detection and segmentation performance[44]. During testing on the remaining 20% split of the data, we intentionally limited postprocessing to binarization and connected-components analysis to ensure an accurate quality assessment of the synthetic datasets. Our primary objective here was to evaluate the raw segmentation capabilities of models trained on these datasets. Postprocessing techniques can compensate for quality gaps in the synthetic data. For example, watershed-based postprocessing can mitigate the issue of overlapping particles that remain connected during testing. Additionally, we benchmark the synthetic data models, except in the AgNW case, against the test performance of a model trained on real data, which serves as a desired performance reference for the synthetic data models. Quantitative results for the three comparable cases are presented in Table 1, while visual segmentation results are shown in Fig. 4. The qualitative visual results for AgNW in SEM are provided separately in Fig. 5. For details on the evaluation procedure, refer to "Workflow for deep learning-based segmentation of nanoparticles".

In the TiO_2 HIM case, the segmentation model trained on real data achieved the best results, closely matching the ground truth with a DSC of $0.968 \pm 3.16 \times 10^{-4}$, AP_{50} of 0.737 ± 0.014, and PQ of $0.938 \pm 5.93 \times 10^{-4}$. Among the models trained on synthetic data, our model outperforms the one trained on synthetic data from Mill et al., achieving a DSC of 0.932 ± 0.003 compared to 0.906 ± 0.009 and a PQ of 0.874 ± 0.005 versus 0.829 ± 0.015, indicating better segmentation accuracy. However, Mill

Table 1 | Quantitative evaluation results of segmentation performance for real and synthetic data

Domain	Model	DSC	AP_{50}	PQ
	Model–Real	**$0.968 \pm 3.16 \times 10^{-4}$**	**0.737 ± 0.014**	**$0.938 \pm 5.93 \times 10^{-4}$**
TiO_2 HIM	Model–Synth Mill et al.	0.906 ± 0.009	0.493 ± 0.020	0.829 ± 0.015
	Model–Synth Ours	0.932 ± 0.003	0.393 ± 0.016	0.874 ± 0.005
	Model–Real	**$0.955 \pm 9.49 \times 10^{-4}$**	**0.945 ± 0.016**	**0.914 ± 0.002**
SiO_2 HIM	Model–Synth Mill et al.	0.786 ± 0.002	0.375 ± 0.004	0.659 ± 0.003
	Model–Synth Ours	$0.860 \pm 4.86 \times 10^{-4}$	0.478 ± 0.011	$0.754 \pm 6.21 \times 10^{-4}$
	Model–Real	**0.964 ± 0.001**	**0.567 ± 0.012**	**0.930 ± 0.001**
TiO_2 SEM	Model–Synth Rühle et al.	0.911 ± 0.001	0.467 ± 0.017	0.837 ± 0.002
	Model–Synth Ours	0.916 ± 0.003	0.474 ± 0.033	0.845 ± 0.004

The table presents the mean and variance of test performance across three runs, measured by the Dice Similarity Coefficient (DSC), Average Precision at 50% IoU (AP_{50}), and Panoptic Quality (PQ) for different segmentation models trained on real and synthetic datasets across various domains: TiO_2 in HIM, SiO_2 in HIM, and TiO_2 in SEM. The "Model–Real" rows represent the averaged test performance of the real-data models. "Model–Synth Mill et al." refers to models trained on synthetic data generated by Mill et al.[28]. Similarly, "Model–Synth Rühle et al." refers to models trained on synthetic data generated by Rühle et al.[23]. "Model–Synth Ours" refers to models trained on synthetic data generated by our DiffRenderGAN approach. Bold values indicate the best scores for each metric within a domain, and underlined values highlight the top scores among synthetic models.

et al.'s model achieved a higher AP_{50} score (0.493 ± 0.020 vs. 0.393 ± 0.016), suggesting better precision in particle localization and separation. Figure 4a shows that the model trained on real data most closely matches the ground truth images. The model trained on Mill et al.'s synthetic data tends to oversegment, introducing frequent false positives, but excels at distinguishing individual instances of nanoparticles. In contrast, our model trained on synthetic data displays false negatives, such as partially unfilled particles, but introduces fewer false positives. However, it struggles with the separation of nanoparticles, particularly with smaller instances, as indicated by the quantitative results.

In the SiO_2 HIM case, the model trained on real data demonstrated superior performance once again, achieving a DSC of $0.955 \pm 9.49 \times 10^{-4}$, AP_{50} of 0.945 ± 0.016, and a PQ of 0.914 ± 0.002. Among the models trained on synthetic data, our synthetic data model performed better than Mill et al.'s approach across all metrics. Our model achieved a higher DSC ($0.860 \pm 4.86 \times 10^{-4}$ vs. 0.786 ± 0.002), AP_{50} (0.478 ± 0.011 vs. 0.375 ± 0.004), and PQ ($0.754 \pm 6.21 \times 10^{-4}$ vs. 0.659 ± 0.003), indicating a better representation of real-world SiO_2 HIM data. Qualitatively (see Fig. 4b), the results from the real data models correspond most closely to the ground truth images. In contrast, our synthetic data model effectively identifies true positives but introduces false positives, particularly in the form of small particle instances. Mill et al.'s synthetic data model struggles with both true positive identification and the avoidance of frequent false positives.

In the TiO_2 SEM domain, among all models, the segmentation model trained on real data achieved the highest performance, with a DSC of 0.964 ± 0.001, a PQ of 0.930 ± 0.001, and an AP_{50} score of 0.567 ± 0.012. The models trained on synthetic data performed very similarly, with our model achieving a DSC of 0.916 ± 0.003 and a PQ of 0.845 ± 0.004, while Rühle et al.'s model reached a DSC of 0.911 ± 0.001 and a PQ of 0.837 ± 0.002. Their AP_{50} score were also closely matched, at 0.474 ± 0.033 for our model and 0.467 ± 0.017 for Rühle et al.'s. These results indicate that both synthetic models offer comparable segmentation accuracy and instance detection, with only marginal variations. The qualitative results in Fig. 4c further illustrate this similarity, showing that both synthetic models effectively segment particles and handle particle separations in a comparable manner. While slight differences in segmentation behavior exist, such as our model's tendency for oversegmentation, both approaches perform nearly equally.

For the AgNW in the SEM case, we conducted a qualitative evaluation due to the absence of annotated ground truth data. Figure 5 presents the segmentation performance of a model trained on synthetic AgNW images generated by DiffRenderGAN. Overall, the model performs well in identifying nanowires, though frequent false negatives can be observed. Despite these errors, the model effectively segments nanowires, highlighting the potential of our synthetic data for this application. Figure 5 also demonstrates an example application of integrating DiffRenderGAN's framework into nanowire image quantification. Specifically, we apply local thickness calculations to examine the rod-like structure of the AgNWs. The figure shows an overlay of local thickness measurements based on the model's segmentations, along with the corresponding thickness distributions.

Discussion

In this study, we present DiffRenderGAN, a novel generative model designed for generating synthetic training data for microscopy analysis. By integrating differentiable rendering into a GAN, DiffRenderGAN directly addresses the challenges posed by the limited availability of (annotated) data, which represents a significant bottleneck in training deep learning models for segmentation and analysis in microscopy research.

Our experiments demonstrate that the method performs well even with limited data, provided that sufficient particle diversity and structural complexity are present. In our study, we utilized datasets ranging from 7 to 10 scans. In this context, we also observed that the balance between generator and discriminator during training can serve as a useful indicator of whether the overall training setup is appropriate. This includes, but is not limited to, aspects such as the suitability and representativeness of the real data.

We assessed DiffRenderGAN across various material morphologies and modalities, including TiO_2 and SiO_2 in HIM, TiO_2 in SEM, and AgNW for rod-like particles in SEM. Our results showed that while models trained on real data consistently achieved the highest performance across segmentation metrics, DiffRenderGAN's synthetic data often matched or exceeded the performance of existing synthetic data techniques. Despite some challenges, particularly in improving particle segregation, DiffRenderGAN shows promise in generating high-quality synthetic data for segmentation-based applications across multimodal microscopy domains, effectively narrowing the domain gap between synthetic and real data. Additionally, it offers a simplified and streamlined approach using a single model for image generation, avoiding multi-stage model training or time-intensive expert-guided rendering methods. This limits manual intervention to providing basic nanoparticle meshes and selecting a scale and positional strategy for realistic nanoparticle mesh alignment.

To further enhance realism, future versions of DiffRenderGAN could incorporate the simulation of scanning-related artifacts such as charging effects, astigmatism, and defocusing, which are commonly observed in SEM and HIM images. Similar to our approach of modeling real-measured noise using a Gaussian distribution, these artifacts could be represented by optimizable known operators[26]. For instance, a trainable Gaussian blur filter could be applied to the output of the differentiable renderer within the generator to approximate blurring effects.

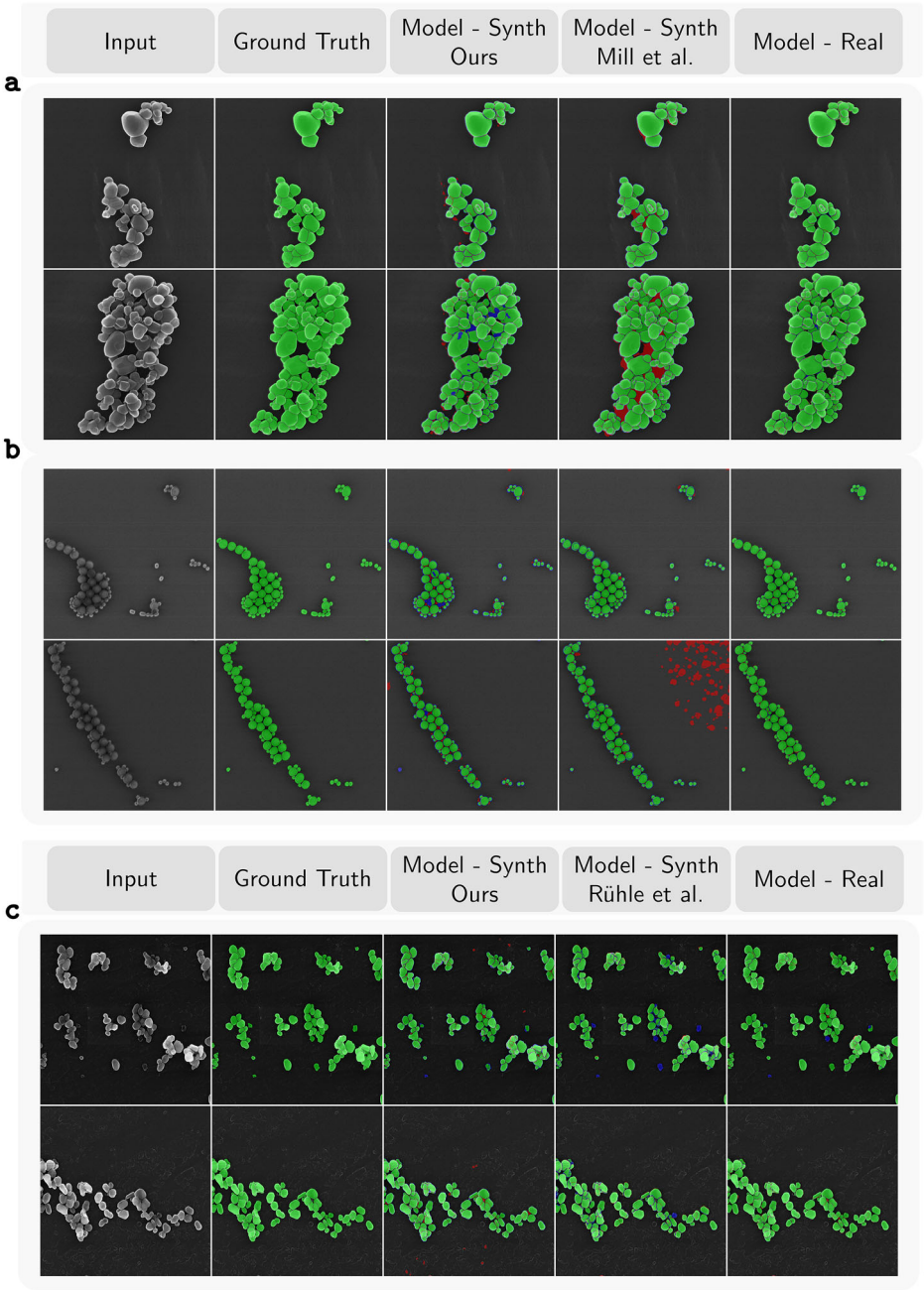

Fig. 4 | Excerpt of segmentation test results from models trained on real and synthetic data. Input images with overlaying corresponding ground truth masks are compared with the segmentation results from models trained on synthetic data generated by our method, synthetic data from a prior study, and a model trained on real data. In comparison with the ground truth masks, green overlays denote true positive pixels, red overlays indicate false positives (pixels incorrectly identified as particles), blue overlays highlight missed particles (false negatives), and pixels without an overlay represent true negatives. Each image pair is selected based on the run with the highest DSC for each model. **a** TiO_2, **b** SiO_2 in HIM from Mill et al.[28]. **c** TiO_2 in SEM from Rühle et al.[23], here cropped for visualization reasons.

Given DiffRenderGAN's limitation of generating arbitrary textures to simulate damaged particles, dust, etc., future research should investigate how to address this problem, as such features are particularly hard to model explicitly in 3D. One possible idea is to extend DiffRenderGAN with a second generator branch focusing on synthesizing complex textures, potentially using a neural network, given its strength in capturing stochastic patterns. The generated textures could then be combined with the results of the differentiable renderer branch, enhancing the realism of the synthetic images.

Looking ahead, DiffRenderGAN's potential applications extend beyond HIM and SEM. Future research should explore its use with other imaging techniques, such as atomic force microscopy, computed tomography, and transmission electron microscopy (TEM), as well as across a broader range of nanomaterials. We demonstrated that although DiffRenderGAN relies on a differentiable renderer library originally designed for light-based interactions, its framework is flexible enough to approximate key features of electron and ion imaging[45]. Moreover, the extensibility of the renderer library used in this work makes it possible to implement custom BSDFs tailored to specific modalities, enabling the simulation of non-light-based contrast mechanisms and transmission-mode responses[45]. While these adaptations do not aim to replicate physically accurate processes, they provide a practical means of generating realistic image data.

In conclusion, DiffRenderGAN represents a substantial advancement in synthetic data generation for microscopy, offering an efficient, scalable, and integrated solution. Although a representativeness gap compared to real data remains, DiffRenderGAN significantly reduces this gap, paving the way for more robust and comprehensive image-based analyses in the study of complex nanomaterial systems.

Methods

Image acquisition and processing

Details regarding sample preparation can be found in the respective publications[5,23,28]. Data for silver nanowires were provided by the authors of

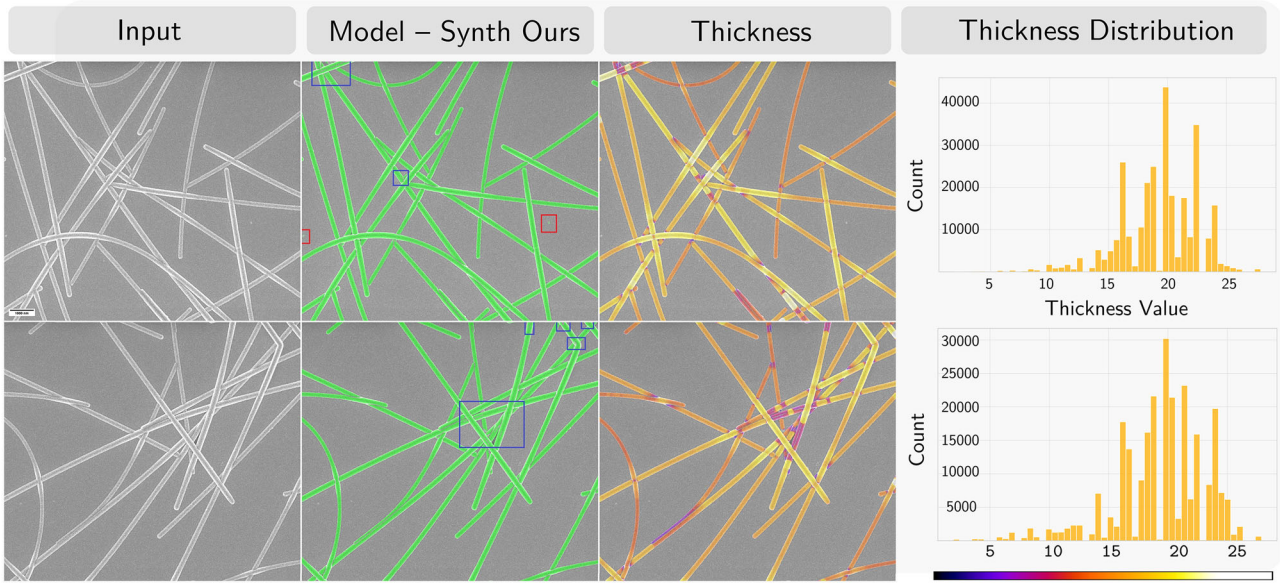

Fig. 5 | Qualitative evaluation of segmentation performance and local thickness estimation for AgNW in SEM. This figure presents a visual analysis of AgNW segmentation and local thickness estimation using a segmentation model trained exclusively on synthetic data generated by the DiffRenderGAN framework. The left column displays two randomly selected SEM images of AgNWs used for testing. The second column shows a green overlay representing the segmentation results, with notable false negatives highlighted in blue and false positives outlined in red. The third column shows a local thickness estimation overlaid as a heatmap based on the segmentation results, where brighter values represent thicker regions. The right column depicts the corresponding local thickness distributions for each scan, providing insights into morphological variations and potential applications in nanotechnology.

ref. 5 (personal communication, unpublished). The sample consisted of AgNWs synthesized according to Korte et al.[46], which were drop-cast onto a silicon substrate and coated with 100 nm of aluminum-doped zinc oxide (AZO) via atomic layer deposition (ALD), an indium-free electrode. Details regarding the sample preparation and the use of this material can be found in the respective publication by Góbelt et al.[5]. The sample was imaged using a Zeiss MultiSEM 505 multibeam SEM at a landing energy of 3 keV. The MultiSEM 505 employs 61 primary electron beams for parallel SEM imaging of large sample surfaces at high resolution[47]. For this sample, the step size was set to 10 nm, and the images were 1252 × 1092 pixels. In total, 10431 individual images (171 locations × 61 beams) with a combined size of more than 14 GPixel and an area of ~1 mm^2 were scanned within less than 5 minutes. From the dataset, twelve images were randomly selected for this study. Of these, ten images were used for training DiffRenderGAN, while the remaining two images were reserved for testing.

Rühle et al. 2021 used TiO2 nanomaterial from the Horizon 2020 project ACEnano, which was ultrasonicated in ultrapure water and drop-cast onto conventional carbon TEM grids and subsequently analyzed using a FEG-SEM (Supra 40, Zeiss) with an In-Lens detector and in transmission mode (not used in DiffRenderGAN evaluation). Real data was sourced from the repository specified in the supplementary section of Rühle et al.[23], comprising 40 SEM images with 1024 × 768 pixels. Out of these, 32 images were used for training our approach and Rühle et al's approach. While eight images were reserved for testing, 1000 annotated synthetic images were generated using the available software from their repository[23]. Each synthetic image had a resolution of 512 × 312 pixels and was created following the original procedure.

In Mill et al., SiO2 nanoparticles with two different diameters and food grade TiO2 nanoparticles (E171) with a size distribution of 20 to 240 nm, both deposited on silicon chips (reference AGAR: G3390-10), were obtained from the "Laboratoire National de métrologie et d'Essais". Secondary electron images of the particles were obtained on a Zeiss ORION NanoFab using the helium-ion beam at an energy of 25 keV and a beam current of 0.5 pA. The NanoFab used a side-mounted secondary electron detector similar to an Everhardt-Thornley type detector. Synthetic datasets included 180 SiO_2 and 180 TiO_2 annotation-paired images at a resolution of 2031 × 2031 pixels. Additionally, eight real TiO_2 (six used for training DiffRenderGAN, two reserved for testing) and nine real SiO_2 (seven used for training DiffRenderGAN, two reserved for testing) annotation-paired images, each with a resolution of 2031 × 2031 pixels, were provided upon request.

Model design

Our GAN model employs a three-layer PatchGAN discriminator based on the CycleGAN architecture[25]. The generator consists of three key modules:

- **Regression model**: A five-layer deep neural network, where the first four layers consist of *Dropout*, a *Fully Connected Layer* (*in* = 128, *out* 113128), and *ReLU* activation. The final layer is a *Fully Connected Layer* with *Sigmoid* activation, responsible for regressing BSDF parameters and the noise scale. *Weight normalization* is applied across all layers.
- **Differentiable rendering function**: Utilizing Mitsuba 3.4 *cuda_ad_mono* Variant[45], this module processes the current virtual scene state and the parameters predicted by the regression model to generate rendered images.
- **Noise-adding function**: After rendering, zero-centered scalable Gaussian noise is added to the images to simulate realistic imaging conditions.

Virtual scene design

Before conducting experiments, a virtual scene (utilized by Mitsuba 3[45]) was designed. This scene consists of a rectangular mesh acting as a stage, a toroidal light source surrounding the stage mesh, and a camera aligned perpendicularly to the center of the stage. An additional rectangular mesh light source is positioned above the camera. Nanoparticle meshes should be centrally located within the torus and stage mesh and are dynamically scaled and translated during training. After each image generation, the nanoparticle meshes reset to their initial position in the center of the stage mesh. The scene is rendered using a *Perspective Sensor*, a *Stratified Sampler*, and a *Gaussian Reconstruction Filter*. Both the stage and the nanoparticle meshes utilize a Principled BSDF. All BSDF parameters not involved in the optimization process remain fixed at their default values. The scene integrator is set to *Direct Reparam*, and the *Area Light* plugin is used for both emitters. The optimized BSDF parameters for the nanoparticle mesh in the

experiments include *Roughness*, *Base Color*, *Sheen*, *Sheen Tint*, and *Specular Tint*, while the stage BSDF optimizes only the *Base Color* parameter. Emission values remain fixed during optimization, with the toroid mesh intensity set to 1.0 and the rectangle mesh intensity set to 0.1.

Nanoparticle mesh modeling

Prior to training DiffRenderGAN, nanomaterial meshes were modeled using Blender version 3.6. Predesigned meshes were adapted to match the morphologies of real particles across different domains: cubes for TiO_2, cones for Ag, and spheres for SiO_2. For each material, a collection of meshes was duplicated and, where necessary, deformed through bending, vertex translation, and rotation.

The following configurations were used in our experiments:

- 10 bent and randomly rotated cones for AgNW SEM
- 20 non-deformed spheres for SiO_2 HIM
- 15 deformed and smoothed cubes for TiO_2 HIM
- 40 deformed and smoothed cubes for TiO_2 SEM

To achieve smoother mesh surfaces, the Blender *Remesh* modifier was applied. Each mesh was positioned at the origin (0, 0, 0) and exported using the Mitsuba-Blender plugin to ensure compatibility with the differentiable rendering software. This workflow was designed to require minimal expertise in 3D rendering.

Transformation computation

To simulate realistic particle sizes and spatial distributions in synthetic images, a size distribution (bimodal, lognormal, or normal) and a spatial arrangement model (random or agglomerated) are selected based on expert analysis of real images prior to training. Once the size and positional distributions are chosen, along with the synthetic image sample size, a transformation tensor is computed. This tensor is generated by sampling from the selected size and positional distributions for each image optimized during training, assigning a scaling factor and positional coordinates to each mesh.

It is critical that the size and spatial distribution model parameters align with the exported sizes of the nanoparticle meshes. Otherwise, synthetic images may contain nanoparticles that are either too large or too small. For random spatial arrangements, the meshes are uniformly distributed within the virtual scene, either in a planar configuration (e.g., AgNW SEM) or in three-dimensional space. For agglomerated arrangements, a Poisson disk-based sampling algorithm was employed to simulate clusters[39].

Model training

All DiffRenderGAN models were trained using PyTorch[48]. The generator's learning rate was set to 0.0002, while the discriminator's learning rate was set to 0.0001, both optimized using Adam[38]. Xavier initialization was applied to both the generator and discriminator[49]. For all experiments, DiffRenderGAN was trained on 256 × 256 pixel image patches for 50 epochs. Each training dataset was cropped into overlapping patches of size 256 × 256 pixels. DiffRenderGAN was then trained on image patches containing at least three fully displayed particles while avoiding repetitive particle patches. Each experiment utilized a batch size of 1. The image patches included in the training were as follows: 82 for AgNW, 56 for SiO_2 in HIM, 126 for TiO_2 in HIM, and 124 for TiO_2 in SEM. As real data augmentation techniques, we employed random flipping, rotation, and transposing. To monitor the quality of synthetic images, after each epoch, a synthetic dataset matching the size of the real dataset was generated, and the FID score was calculated. The best epoch was selected as detailed in the main text.

Model inference

After training, the generator was loaded with the respective best epoch state. During runtime, an additional duplicate scene, without the stage mesh, was created, where an *AOV* integrator was used. This integrator enables the rendering of labeled images displaying unique identifiers for each mesh observed in the camera's field of view, necessitating the removal of the stage mesh. The generated label images were processed through rounding to ensure discrete label values. Subsequently, these label images were binarized, and an additional contour class was added. For TiO_2 and SiO_2 in HIM and AgNW in SEM, a contour thickness of four pixels was used, while for TiO_2 in SEM, a thickness of one pixel was applied. To ensure meaningful synthetic images during inference, only images where particles exhibited sufficient contrast against the background were rendered. Specifically, based on the mask information, we automatically removed images during inference where the mean intensity of the particles was less than 15% in comparison to the mean intensity of the background. Following this automated strategy during inference, each experiment produced 1000 paired synthetic images with their respective annotated masks.

Workflow for deep learning-based segmentation of nanoparticles

For performance comparisons, we employed the nnUNet framework[50], which automatically configures model parameters based on the characteristics of each individual dataset. This approach eliminates potential performance bias that could arise from manual model selection and configuration. All segmentation models were trained for multiclass segmentation using nnUNet's default training procedure, classifying pixels into three categories: particle, contour, and background. The contour class specifically aids in distinguishing overlapping particles. During the addition of the contour class for real data and synthetic data from other methods, we ensured that no particle information in the respective masks was overwritten by the addition of contour class pixels.

To ensure robust evaluation of our models and mitigate potential biases introduced by artifact-rich environments in original scans, we supplemented our synthetic image datasets with 200 real background patches (i.e., images without particles) randomly sampled from overlapping patches of the corresponding real training data, used during DiffRenderGAN training. The rationale behind this supplementation was to encourage our synthetic data models to accurately distinguish true particles from irrelevant artifacts, such as dirt or preparational anomalies, during segmentation tasks. Since our proposed method does not provide additional meshes for artifacts and only generates "clean" images, this strategy introduces additional robustness under varying imaging conditions. We note that this aspect is explicitly or implicitly considered across all approaches: Mill et al. supplemented their data by including dirt textures as synthetic backgrounds, partially addressing artifact-related challenges. Rühle et al.'s GAN-based approach inherently integrates the generation of background and artifacts as long as they are present in the training dataset. Therefore, we follow these methods as proposed by the original authors.

The datasets used for nnUNet training included the following:

- **TiO_2HIM**: Our segmentation model was trained on our synthetic dataset (1000 images, 256 × 256 pixels) supplemented with real background patches (200 images, 256 × 256 pixels; total: 1200 images). Mill et al.'s model was trained on their synthetic dataset (180 images, 2031 × 2031 pixels). The real-data model was trained on 294, 256 × 256 image patches extracted from six real images in the training split, utilizing the available respective ground truth masks for training.
- **SiO_2HIM**: Our segmentation model was trained on our synthetic dataset (1000 images, 256 × 256 pixels) supplemented with real background patches (200 images, 256 × 256 pixels; total: 1200 images). Mill et al.'s model was trained on their synthetic dataset (180 images, 2031 × 2031 pixels). The real-data model was trained on 343, 256 × 256 image patches extracted from seven real images in the training split, utilizing the available respective ground truth masks for training.
- **TiO_2SEM**: Our segmentation model was trained on our synthetic dataset (1000 images, 256 × 256 pixels) supplemented with real background patches (200 images, 256 × 256 pixels; total: 1200 images) and tested on eight real images. Rühle et al.'s model, trained on their synthetic dataset (1000 images, 512 × 312 pixels). The real-data model was trained on 256, 256 × 256 image patches extracted from 32 real images in the training split, utilizing the available respective ground truth masks for training.

- **AgNW SEM**: Our segmentation model was trained on our synthetic dataset (1000 images, 256 × 256 pixels) supplemented with real background patches (200 images, 256 × 256 pixels; total: 1200 images).

Each model was trained for three runs. After training, each model from each run within its respective domain was tested on the same real test data, using a step size of 0.1 (TiO_2 HIM: two image scans; SiO_2 HIM: two image scans; TiO_2 SEM: eight image scans). After testing, each image was binarized using only the particle class information. We then computed the mean test performance for each model within each run and calculated the mean and variance of the test performance across all runs using the evaluation metrics introduced in "Deep learning-based segmentation of nanoparticles trained on synthetic images".

Technical notes

All experiments involving DiffRenderGAN and nnUNet were conducted using Python 3.9.18 on an NVIDIA A40 GPU with CUDA 12.3.

Data availability

Real and synthetic HIM images reported by Mill et al., as well as real SEM images from Göbelt et al., can be obtained from the corresponding authors of the respective studies upon request. The SEM dataset introduced by Rühle et al. can be accessed through the repository detailed in the respective supplementary information. The synthetic images used in this study were produced using the original code provided by the authors in their publicly accessible repository.

Code availability

The DiffRenderGAN source code, including fully trained model weights for each case, is publicly available at GitHub. All necessary dependencies and detailed usage instructions are provided in the repository documentation.

Received: 1 April 2025; Accepted: 13 June 2025;
Published online: 01 July 2025

References

1. Wu, A. & Ren, W. *TiO2 nanoparticles: applications in nanobiotechnology and nanomedicine*. (John Wiley & Sons, Weinheim, 2020).
2. Huang, Y. et al. Silica nanoparticles: Biomedical applications and toxicity. *Biomed. Pharmacother.* **151**, 113053 (2022).
3. Gupta, S. M. & Tripathi, M. A review of tio 2 nanoparticles. *Chin. Sci. Bull.* **56**, 1639–1657 (2011).
4. Nayl, A., Abd-Elhamid, A., Aly, A. A. & Bräse, S. Recent progress in the applications of silica-based nanoparticles. *RSC Adv.* **12**, 13706–13726 (2022).
5. Goebelt, M. et al. Encapsulation of silver nanowire networks by atomic layer deposition for indium-free transparent electrodes. *Nano Energy* **16**, 196–206 (2015).
6. Han, J. et al. Fully indium-free flexible ag nanowires/zno: F composite transparent conductive electrodes with high haze. *J. Mater. Chem. A* **3**, 5375–5384 (2015).
7. Radmilović, V. V. et al. Low temperature solid-state wetting and formation of nanowelds in silver nanowires. *Nanotechnology* **28**, 385701 (2017).
8. Schrenker, N. et al. Mechanical and electrical failure of silver nanowire electrodes: a scale bridging in situ electron microscopy study. *Microsc. Microanalysis* **25**, 2038–2039 (2019).
9. Schrenker, N. J. et al. Microscopic deformation modes and impact of network anisotropy on the mechanical and electrical performance of five-fold twinned silver nanowire electrodes. *ACS Nano* **15**, 362–376 (2020).
10. Bashouti, M. Y. et al. functionalization of silver nanowires surface using ag–c bonds in a sequential reductive method. *ACS Appl. Mater. interfaces* **7**, 21657–21661 (2015).
11. Ronneberger, O., Fischer, P. & Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Navab, N., Hornegger, J., Wells, W. M. & Frangi, A. F. (eds.) *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015*, 234-241 (Springer International Publishing, Cham, 2015).
12. Van der Laak, J., Litjens, G. & Ciompi, F. Deep learning in histopathology: the path to the clinic. *Nat. Med.* **27**, 775–784 (2021).
13. Aubreville, M. et al. Mitosis domain generalization challenge 2022 https://doi.org/10.5281/zenodo.6362337 (2022).
14. Fu, T. et al. Deep-learning-enabled crack detection and analysis in commercial lithium-ion battery cathodes. *Adv. Funct. Mater.* **32**, 2203070 (2022).
15. Shammaa, M. H., Ohtake, Y. & Suzuki, H. Segmentation of multi-material ct data of mechanical parts for extracting boundary surfaces. *Computer-Aided Des.* **42**, 118–128 (2010).
16. Zhang, C., Bengio, S., Hardt, M., Recht, B. & Vinyals, O. Understanding deep learning (still) requires rethinking generalization. *Commun. ACM* **64**, 107–115 (2021).
17. Bishop, C. M. Pattern recognition and machine learning. *Springe google Sch.* **2**, 1122–1128 (2006).
18. Theodoridis, S. & Koutroumbas, K. *Pattern recognition*. (Elsevier, San Diego, 2006).
19. Goodfellow, I. et al. Generative adversarial nets. Advances in neural information processing systems 27 (2014).
20. Frid-Adar, M., Klang, E., Amitai, M., Goldberger, J. & Greenspan, H. Synthetic data augmentation using gan for improved liver lesion classification. In *2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018)*, 289-293 (IEEE, 2018).
21. Jordon, J., Yoon, J. & Van Der Schaar, M. Pate-gan: Generating synthetic data with differential privacy guarantees. In *International conference on learning representations* (2018).
22. Guan, S. & Loew, M. Breast cancer detection using synthetic mammograms from generative adversarial networks in convolutional neural networks. *J. Med. Imaging* **6**, 031411–031411 (2019).
23. Rühle, B., Krumrey, J. F. & Hodoroaba, V.-D. Workflow towards automated segmentation of agglomerated, non-spherical particles from electron microscopy images using artificial neural networks. *Sci. Rep.* **11**, 4942 (2021).
24. Arjovsky, M., Chintala, S. & Bottou, L. Wasserstein generative adversarial networks. In Precup, D. & Teh, Y. W. (eds.) *Proceedings of the 34th International Conference on Machine Learning*, vol. 70 of *Proceedings of Machine Learning Research*, 214-223 (PMLR, Sydney, 2017).
25. Zhu, J.-Y., Park, T., Isola, P. & Efros, A. A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Computer Vision (ICCV), 2017 IEEE International Conference on* (2017).
26. Maier, A. K. et al. Learning with known operators reduces maximum error bounds. *Nat. Mach. Intell.* **1**, 373–380 (2019).
27. Wood, E. et al. Fake it till you make it: Face analysis in the wild using synthetic data alone. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 3681-3691 (2021).
28. Mill, L. et al. Synthetic image rendering solves annotation problem in deep learning nanoparticle segmentation. *Small Methods* **5**, 2100223 (2021).
29. Jakob, W., Speierer, S., Roussel, N. & Vicini, D. Dr.jit: A just-in-time compiler for differentiable rendering. Transactions on Graphics (Proceedings of SIGGRAPH) 41 (2022).
30. Pharr, M., Jakob, W. & Humphreys, G.Physically based rendering: From theory to implementation (MIT Press, Cambridge, 2023).
31. Kajiya, J. T. The rendering equation. In *Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, 143-150 (1986).
32. Rozantsev, A., Lepetit, V. & Fua, P. On rendering synthetic images for training an object detector. *Computer Vis. Image Underst.* **137**, 24–37 (2015).

33. Yoo, T. K., Choi, J. Y. & Kim, H. K. Cyclegan-based deep learning technique for artifact reduction in fundus photography. *Graefe's. Arch. Clin. Exp. Ophthalmol.* **258**, 1631–1637 (2020).
34. Osakabe, T., Tanaka, M., Kinoshita, Y. & Kiya, H. Cyclegan without checkerboard artifacts for counter-forensics of fake-image detection. In *International Workshop on Advanced Imaging Technology (IWAIT) 2021*, vol. 11766, 51-55 (SPIE, 2021).
35. de Bel, T., Bokhorst, J.-M., van der Laak, J. & Litjens, G. Residual cyclegan for robust domain transformation of histopathological tissue slides. *Med. Image Anal.* **70**, 102004 (2021).
36. Loubet, G., Holzschuch, N. & Jakob, W. Reparameterizing discontinuous integrands for differentiable rendering. Transactions on Graphics (Proceedings of SIGGRAPH Asia) 38 (2019).
37. Robbins, H. & Monro, S. A stochastic approximation method. The annals of mathematical statistics400-407 (1951).
38. Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization. CoRR abs/1412.6980 https://api.semanticscholar.org/CorpusID:6628106 (2014).
39. Bridson, R. Fast poisson disk sampling in arbitrary dimensions. *SIGGRAPH sketches* **10**, 1 (2007).
40. Rumelhart, D. E., Hinton, G. E. & Williams, R. J. Learning representations by back-propagating errors. *nature* **323**, 533–536 (1986).
41. Parmar, G., Zhang, R. & Zhu, J.-Y. On aliased resizing and surprising subtleties in gan evaluation. In *CVPR* (2022).
42. Müller, D., Soto-Rey, I. & Kramer, F. Towards a guideline for evaluation metrics in medical image segmentation. *BMC Res. Notes* **15**, 210 (2022).
43. Lin, T.-Y. et al. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, 740-755 (Springer, 2014).
44. Kirillov, A., He, K., Girshick, R., Rother, C. & Dollár, P. Panoptic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9404-9413 (2019).
45. Jakob, W. et al. Mitsuba 3 renderer. https://mitsuba-renderer.org. (2022)
46. Korte, K. E., Skrabalak, S. E. & Xia, Y. Rapid synthesis of silver nanowires through a cucl-or cucl 2-mediated polyol process. *J. Mater. Chem.* **18**, 437–441 (2008).
47. Eberle, A. et al. High-resolution, high-throughput imaging with a multibeam scanning electron microscope. *J. Microsc.* **259**, 114–120 (2015).
48. Paszke, A. et al. Pytorch: an imperative style, high-performance deep learning library. *Adv. Neural Inf. Process. Syst.* **32**, https://arxiv.org/abs/1912.01703 (2019).
49. Glorot, X. & Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 249-256 (JMLR Workshop and Conference Proceedings, 2010).
50. Isensee, F., Jaeger, P. F., Kohl, S. A., Petersen, J. & Maier-Hein, K. H. nnu-net: a self-configuring method for deep learning-based biomedical image segmentation. *Nat. Methods* **18**, 203–211 (2021).

Acknowledgements

S.C. was supported by the European Union's H2020 research and innovation program under the Marie Sklodowska-Curie grant agreement AIMed ID: 861138. D.P., D.A., G.S. and S.C. acknowledge the financial support from the European Union within the research projects 4D + nano-SCOPE ID: 810316, LRI ID: C10, STOP ID: 101057961, from the German Research Foundation (DFG) within the research project UNPLOK ID: 523847126, and from the "Freistaat Bayern" and European Union within the project Analytiktechnikum für Gesundheits- und Umweltforschung AGEUM, StMWi-43-6623-22/1/3.

Author contributions

D.P. conceptualization, methodology, software, data curation, validation and analysis, writing—original draft, visualization, writing—review & editing; L.M. & M.H. conceptualization, methodology, software, writing—review & editing; F.V. data curation, methodology, validation and analysis, writing—review & editing; T.H. and P.S. software, writing—review & editing; J.U., D.A., M.T., M.G., F.W. validation and analysis, investigation, writing—review & editing; G.S. data curation, validation and analysis, supervision, writing—review & editing; S.C. data curation, funding acquisition, supervision, writing—review & editing; K.B. conceptualization, methodology, validation, and analysis, supervision, writing—review & editing.

Funding

Open Access funding enabled and organized by Projekt DEAL.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41524-025-01702-6.

Correspondence and requests for materials should be addressed to Dennis Possart.