

# SUPER-RESOLVING 3D NANOSTRUCTURES USING ARTIFICIALLY GENERATED IMAGE DATA AND SPATIAL TRANSPORT SIMULATIONS

ORKUN FURAT<sup>1,\*</sup>, PHILLIP GRÄFENSTEINER<sup>1,\*</sup>, RISHABH SAXENA<sup>2</sup>,  
MARKUS OSENBERG<sup>3</sup>, MATTHIAS NEUMANN<sup>4</sup>, INGO MANKE<sup>3</sup>, THOMAS CARRARO<sup>2</sup>,  
VOLKER SCHMIDT<sup>1</sup>

<sup>1</sup>*Institute of Stochastics, Ulm University, Helmholtzstraße 18, 89069 Ulm, Germany*

<sup>2</sup>*Institute for Modeling and Computational Science, Faculty of Mechanical and Civil Engineering,  
Helmut Schmidt University, Holstenhofweg 85, 22043 Hamburg, Germany*

<sup>3</sup>*Institute of Applied Materials, Helmholtz-Zentrum Berlin für Materialien und Energie,  
Hahn-Meitner-Platz 1, 14109 Berlin, Germany*

<sup>4</sup>*Institute of Statistics, Graz University of Technology, Kopernikusgasse 24/III, 8010 Graz, Austria*

*\*OF and PG contributed equally to this work*

**ABSTRACT.** An approach for improving the performance of super-resolution networks by integrating spatially resolved transport simulations is investigated for three different scaling factors  $\alpha \in \{2, 4, 8\}$ . In addition, the issue of scarcity in training data is addressed by training the networks only on artificial image data, generated by means of a stochastic 3D model that produces digital twins of the nanoporous inner structure of active particles in battery cathodes. The performance of the trained networks is evaluated based on real tomographic image data, and quantified with respect to various geometric descriptors and effective transport properties. It turned out that the integration of transport simulations into the training of super-resolution networks showed an increase in performance for the scaling factors  $\alpha \in \{2, 4\}$ , but a decrease in performance for  $\alpha = 8$ . However, training the networks on artificial image data was effective in all cases.

## 1. INTRODUCTION

Predicting effective properties of functional materials on a nanostructural level is a highly complex task that requires detailed knowledge on the morphology of the specimen in question [1]. In order to obtain this knowledge, it is often necessary to acquire 3D image data at a resolution that is high enough to resolve all relevant details of the nanostructure. On the other hand, the acquired data needs to cover a representative volume element in order to ensure reliability and reproducibility of any subsequent analysis. This highlights the fundamental trade-off between resolution and field-of-view in many imaging techniques.

One approach to combat the need of experimentally acquiring highly-resolved image data is through super-resolution techniques, which artificially up-sample a given image from a lower to a higher resolution. Since imaging at a lower resolution typically is accompanied with a larger field-of-view, super-resolution enables us to measure a larger volume at a lower resolution during the imaging stage, while regaining some of the forgone resolution in post-processing. Within the field of machine learning, convolutional neural networks (CNNs) have been thoroughly studied for super-resolution tasks and have been shown to be an efficient technique superior to many classical methods [2, 3]. However, as neural networks require an extensive database to be trained, such an approach requires a large amount of already available highly-resolved image data.

---

*Key words and phrases.* Super-resolution, stochastic 3D modeling, digital twin, porous medium, FIB-SEM, neural network.

In this paper we attempt to solve the scarcity in training data by application of a stochastic 3D model [4–6] that allows for a computationally cost-effective generation of digital twins, *i.e.*, an artificial specimen of a material that is statistically similar to a real world sample with respect to its geometry, morphology and functional properties [7]. In this way, new image data can be (artificially) generated and subsequently down-sampled in order to create a database of low- and high-resolution pairs of simulated 3D images for the training of the super-resolution CNN. Regarding similar papers where data generation via stochastic modeling and machine learning has been combined, we refer to [8–11].

We study this approach by means of the stochastic 3D model proposed in [12]. More precisely, using an excursion set of a  $\chi^2$ -random field, we generate digital twins of nanoporous particles in cathodes of lithium-ion batteries. These particles exhibit a porous structure on the nanoscale and have been measured by focused ion beam scanning electron microscopy (FIB-SEM) tomography [12]. Based on this tomographic image data, the model was calibrated and also validated by comparing the values of various geometric descriptors computed for model realizations and tomographic image data, respectively [12]. In the present paper, the database generated in this way will be used to train different CNNs performing super-resolution tasks. While this approach still fundamentally requires tomographic image data for the calibration of the stochastic 3D model, the amount of data needed for this is significantly less than that required for the training of a CNN performing super-resolution tasks. Moreover, since the stochastic 3D model is fully parametric, it is possible to vary the morphology of the simulated nanostructures in a systematic way, thereby creating a more comprehensive database for training neural networks.

A primary motivation for obtaining high-resolution 3D image data is their crucial role in supporting numerical simulations of various physical and chemical transport phenomena. Such data serve as a detailed basis for modeling transport processes in porous materials, including fluid flow, diffusion, and reactive transport. In the present paper, we focus specifically on transport processes in battery electrodes. In this context, highly accurate nanostructure reconstructions are essential for the accurate simulation of complex electrochemical processes, where the critical interaction between nanostructure, ion transport, and electrochemical reactions determines the overall performance [13–20]. To ensure that artificially super-resolved image data provide a reliable foundation for numerical simulations, it is crucial that the transport pathways through the material are realistically reproduced. In three-dimensional porous structures, transport often localizes around a few paths that contribute to a major portion of the flow. If the network misclassifies some voxels in such a way that those paths can no longer be used, the resulting error in representing the morphology is larger than the errors conceivable by a voxel-wise comparison between super resolution and ground truth. Thus, in a second study of this paper, we aim to provide the CNN with a weighting of the voxels based on their contribution to the overall flow, thereby guiding the network towards important bottlenecks or connections within the transporting phase. This is done by use of numerical solutions of diffusion problems on the low-resolution nanostructures. In particular, the diffusion problem is solved on low-resolution images of both solid and pore phases. Then, the solutions are provided to the network in a discretized form as two 3D gray scale images in order to perform super-resolution of the low-resolution image of the nanostructure. On low-resolution images, such a solution can be determined within reasonable time.

We compare the performance of CNNs trained only on simulated 3D binary images with that of CNNs trained with the additional information of the diffusion problem at three different scaling factors  $\alpha \in \{2, 4, 8\}$ . As a baseline reference, we also consider upsampling by tri-quadratic interpolation [21]. The trained CNNs are validated by assessing the quality of the super-resolution task on both virtual and experimentally measured image data. For this, we consider metrics such as a simple accuracy rating acquired by counting the correctly classified voxels in the artificially super-resolved image data. As super-resolving large areas of the same phase is typically easy, we also consider the surface accuracy, which only counts correctly classified voxels near the boundary of the solid and pore phase. Moreover, we assess the quality of super-resolution results by means of the transport-related notion of tortuosity, which aims to quantify the degree of winding in transport paths through a given phase of the material. While there are many different definitions of tortuosity [22], we focus on the mean geodesic tortuosity and the effective tortuosity, which correspond to a purely geometric and a physical approach to define tortuosity, respectively. The mean geodesic tortuosity quantifies the average relative length of shortest paths starting from one side of a cuboidal volume element of the material to the opposite side within a given phase. The effective tortuosity emerges from the solution of the diffusion problem. It is defined

via the ratio of the effective and intrinsic diffusion coefficients multiplied by the volume fraction, and quantifies the impact of the morphology on the overall flux in the medium. In this way, we can quantify whether the additional information on the solution of the diffusion problem aids the CNNs in correctly classifying voxels that belong to crucial transport paths in the medium.

The remainder of this paper is organized as follows. In Section 2.1 we present the database considered in this paper. This includes binary tomographic 3D image data acquired by FIB-SEM measurements, corresponding (simulated) binary 3D image data drawn from the calibrated stochastic 3D models, and finally 3D gray scale images representing the solution of the diffusion problem on the solid and pore phase of the nanostructures. Section 2.2 introduces the neural networks used for the super-resolution task. The metrics used for validation of the performance of these networks are described in Section 2.3. A detailed discussion of the results obtained in this study is given in Section 3. Finally, Section 4 concludes.

## 2. METHODS

In this paper we apply super-resolution techniques to tomographic image data depicting the nanostructure of battery electrodes. However, the method investigated here is not limited to battery applications. It can be extended to super-resolution of porous materials with two or more material phases segmented from tomographic data. In particular, it can be applied to materials used in other energy storage and conversion technologies, such as fuel cells, proton exchange membranes, supercapacitors and electrochemical catalysts. In all of these applications, an in-depth understanding of the nano- and microstructures of the considered materials is beneficial.

**2.1. Generating experimentally measured and simulated image data.** This section presents methods for generating the database used in this study, consisting of experimentally measured tomographic image data, as well as artificially generated image data drawn from a stochastic 3D model, and 3D gray scale images representing the solutions of the diffusion problem on the solid and pore phases of the nanostructures.

**2.1.1. Tomographic image data.** In the present paper, we deal with tomographic image data of functional materials, namely active material for lithium-ion batteries. In particular, we focus on the nanostructured porous secondary particles of a battery cathode made of nickel manganese cobalt oxide (NMC111), which have been experimentally studied, *e.g.*, in [23, 24], see Figure 1.

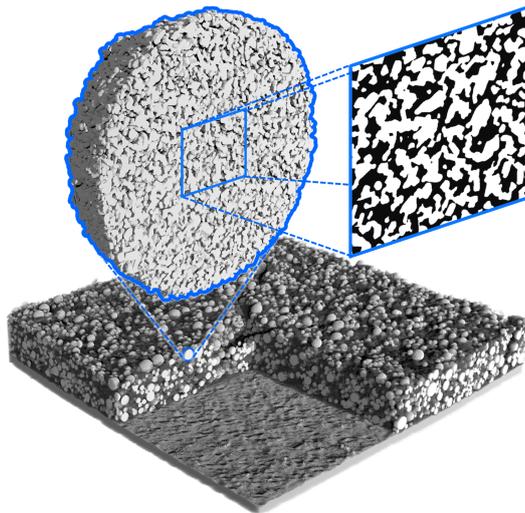


FIGURE 1. Lithium-ion battery cathode with hierarchically structured NMC111 active material particles. The tomographic FIB/SEM image data depicts the inner structure of secondary particles, visualized the blue box, where the solid phase is indicated in white, and the pore phase in black.

The choice of this material is motivated by the availability of a well-established stochastic 3D model [12] for the generation of virtual material morphologies, which enables the generation of large data sets required for training super-resolution networks. Synthetic data plays a crucial role in complementing real experimental data and allows the neural networks to generalize effectively over a range of nanostructural patterns.

The material used in this study comprises five different samples of hierarchically structured NMC111 cathodes, each with different morphologies characterized by various geometric descriptors. These samples consist of porous secondary particles, which are themselves composed of smaller primary particles. More precisely, secondary particles have been produced by sintering primary particles—a process in which the material is heated to just below the melting point to bond the primary particles and change the inner nanostructure of secondary particles. To investigate the effects of particle size and sintering conditions, three different primary particle sizes were considered, classified as fine, medium and coarse. In addition, two different sintering temperatures, 850°C and 900°C, were applied, which further influence the inner morphology of secondary particles. To capture the nanostructure, all samples have been analyzed using 3D FIB-SEM with a resolution of 10 nm/voxel, resulting in tomographic image data for the inner structure of the secondary particles as shown in Figure 1. The size of each measured image is  $416^3$  voxels. The obtained gray scale images were binarized using the Otsu thresholding method as described in [12]. For more details on the material as well as on the acquisition and processing of 3D image data, the reader is referred to [12].

**2.1.2. Stochastic 3D model for porous nanostructures.** Based on the tomographic image data obtained by FIB-SEM measurements, as previously described, a stochastic 3D model for the solid phase of hierarchically structured NMC111 particles has been developed, which effectively captures the inner structure of the material, as described in [12]. The model is based on excursion sets of random fields, using tools from stochastic geometry [4] and mathematical morphology [5]. Note that excursion sets of random fields have recently also been used to model the nanostructure of hierarchically structured electrodes for sodium-ion batteries [25].

The stochastic 3D nanostructure model was validated with respect to various geometrical descriptors that were not used for model fitting, such as mean geodesic tortuosity, which quantifies shortest path lengths within a certain phase, or constrictivity, which measures the strength of bottleneck effects within a certain phase. In addition, the so-called M-factor was considered, which is the ratio of effective to intrinsic conductivity in the case of the solid phase, and the ratio of effective to intrinsic diffusivity in the case of the pore phase. In [12] this descriptor of effective (charge or mass) transport was numerically computed for both model realizations and tomographic image data of measured active material samples by means of a finite element method [26, 27]. The validation procedure performed in [12] showed a high agreement for all considered metrics across all samples, which suggests that the model realizations provide a suitable database for the training of the super-resolution network introduced in Section 2.2 below.

To make the present paper more self-contained, we briefly recall how the stochastic 3D model introduced in [12] is defined. Let  $X_1$  and  $X_2$  be two independent copies of a motion-invariant, *i.e.*, stationary and isotropic, Gaussian random field  $X = \{X(t) : t \in \mathbb{R}^3\}$ , the expectation function  $\mathbb{E}X : \mathbb{R}^3 \rightarrow \mathbb{R}$  and variance function  $\text{Var } X : \mathbb{R}^3 \rightarrow \mathbb{R}$  of which fulfill  $\mathbb{E}X(t) = 0$  and  $\text{Var } X(t) = 1$  for all  $t \in \mathbb{R}^3$ . Under the above-mentioned assumptions, the distribution of such a random field  $X$  is uniquely determined by its covariance function  $\rho_X : [0, \infty) \rightarrow \mathbb{R}$ , given by  $\rho_X(h) = \text{Cov}(X(s), X(t))$  for each  $h > 0$ , where  $s, t \in \mathbb{R}^3$  with  $|s - t| = h$ . Due to the assumption of motion invariance, the definition of  $\rho_X(h)$  does not depend on the specific choice of  $s$  and  $t$ , but just on distance  $|s - t| = h$  between them. Furthermore, let  $Y = \{Y(t) : t \in \mathbb{R}^3\}$ , where  $Y(t) = X_1^2(t) + X_2^2(t)$  for each  $t \in \mathbb{R}^3$ . The random field  $Y$  is then called a  $\chi^2$ -field with two degrees of freedom. Similar to the Gaussian random field  $X$ , the distribution of such a field  $Y$  is uniquely determined by the covariance function  $\rho_X$ . The solid phase of the nanostructured active material observed in tomographic image data is now modeled by the random excursion set  $\Xi = \{t \in \mathbb{R}^3 : Y(t) \geq \nu\}$ , for some  $\nu > 0$ , which is the set of locations  $t \in \mathbb{R}^3$  at which the value  $Y(t)$  of the random field  $Y$  exceeds a given threshold  $\nu$ . In order to calibrate the model, the covariance function  $\rho_X$  and the parameter  $\nu$ , which is directly connected to the volume fraction of the random set  $\Xi$ , need to be estimated from tomographic image data. For estimating  $\rho_X$ , a parametric

fit of the form  $\rho(h) = \exp(-\gamma^2 h)$ ,  $h \geq 0$ , for some  $\gamma > 0$ , was shown to be appropriate. The model is therefore characterized to the two parameters  $\nu$  and  $\gamma$ . For each of the five samples measured by FIB-SEM, the value of the parameter pair  $(\nu, \gamma)$  is estimated from tomographic image data. A more detailed description of the estimation of  $\nu$  and  $\gamma$  is given in [12].

**2.1.3. Generating a database of low- and high-resolution image data.** First, we introduce some basic notation for 3D images defined on a cuboidal grid, which will be used throughout this paper. For some fixed integers  $m_1, m_2, m_3 \in \mathbb{N} = \{1, 2, \dots\}$ , let

$$W = \{1, \dots, m_1\} \times \{1, \dots, m_2\} \times \{1, \dots, m_3\}. \quad (1)$$

Moreover, in the context of the super-resolution tasks considered in this paper, we will need to change the resolution of the grid by a scaling factor  $\alpha \in \{2, 4, 8\}$ . Therefore, we also consider the domain

$$W_\alpha = \{1, \dots, \alpha m_1\} \times \{1, \dots, \alpha m_2\} \times \{1, \dots, \alpha m_3\}. \quad (2)$$

As a database for training neural networks to perform super-resolution, we draw realizations from the excursion set model  $\Xi$  described in Section 2.1.2. More precisely, for each of the five experimentally measured samples, 10 realizations are drawn from the corresponding model, using the respective pairs  $(\nu, \gamma)$  of parameters estimated from tomographic image data. The sampling window for these (high-resolution) images consists of  $800^3$  voxels, *i.e.*,  $W_8 = \{1, \dots, 800\}^3$ . In other words, a model realization corresponds to a binary 3D image  $I: \{1, \dots, 800\}^3 \rightarrow \{0, 1\}$ , where the values 0 and 1 indicate that a voxel belongs to the pore phase and the solid phase of the active material, respectively. The resolution of these binary images coincides with the resolution of the tomographic image data which has been used for the calibration of model parameters, *i.e.*, the resolution is 10 nm/voxel. These realizations constitute the database of virtual high-resolution nanostructures and are denoted by  $I_{800}^{s,i}$  for each experimentally measured scenario  $s \in \{1, \dots, 5\}$  and for each statistically equivalent model realization  $i \in \{1, \dots, 10\}$ . Note that for training a neural network to perform super-resolution, we will require pairs of low- and high-resolution image data of the same morphologies. Therefore, in order to generate virtual low-resolution image data of the nanostructure, we down-sample the high-resolution image data by deleting every second slice along each of the three major axes, see Figure 2.

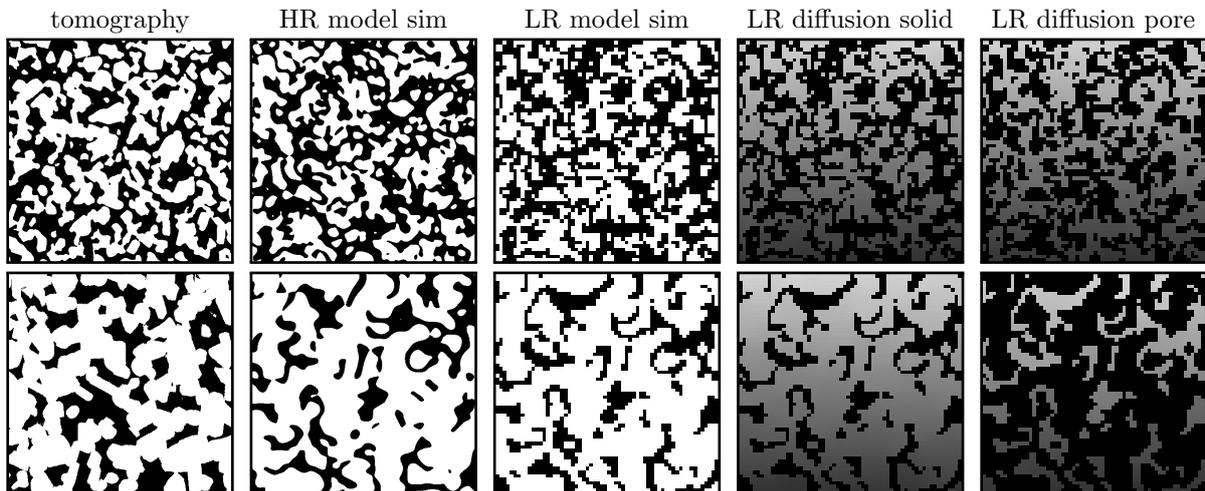


FIGURE 2. 2D slices of image data for two different samples: Medium-grained primary particles that have been sintered at 850 °C (top row), and coarse-grained primary particles sintered at 900 °C (bottom row). Columns from left to right: Tomographic image data, high-resolution (HR) model realization, low-resolution (LR) model realization, LR simulation of diffusive transport in the solid phase, LR simulation of diffusive transport in the pore phase. For the binary images in the first three columns, the solid phase is indicated in white. In the remaining two columns, the gray scale values of individual voxels indicate the value of the solution to the diffusion problem described in Section 2.1.4.

More precisely, the down-sampled image  $I_{400}^{s,i}: \{1, \dots, 400\}^3 \rightarrow \{0, 1\}$  of the high-resolution image  $I_{800}^{s,i}: \{1, \dots, 800\}^3 \rightarrow \{0, 1\}$  is given by

$$I_{400}^{s,i}(j, k, l) = I_{800}^i(2j - 1, 2k - 1, 2l - 1), \quad (3)$$

for any  $j, k, l \in \{1, \dots, 400\}$ ,  $i \in \{1, \dots, 10\}$  and  $s \in \{1, \dots, 5\}$ . In this way, we obtain a coarser version of each realization with  $400^3$  voxels, with a resolution of 20 nm/voxel. Furthermore, the procedure described in Eq. (3) is repeated twice, resulting in low-resolution images  $I_{200}^{s,i}$  and  $I_{100}^{s,i}$ ,  $i = 1, \dots, 10$ ,  $s = 1, \dots, 5$ , with  $200^3$  and  $100^3$  voxels, and corresponding resolutions of 40 nm/voxel and 80 nm/voxel, respectively. This method of direct downsampling is often used in super-resolution studies to generate a database of low- and high-resolution image pairs [28–33]. As we are only interested in the super-resolution of binary images, we chose to omit the commonly used blurring step before the downsampling, so that no additional thresholding is necessary.

*2.1.4. Numerical simulation of diffusive transport.* To enhance the performance of the CNNs, which will be introduced in Section 2.2 below for super-resolving image data of nanostructures, additional information is provided to the networks through 3D transport simulations. These simulations capture crucial physical properties of the nanostructure and are expected to offer valuable insights into voxel connectivity. As the resolution decreases, certain transport paths may become misrepresented because of voxels being assigned to only one material phase, although in reality both phases might be present in finer resolutions within the volume of the corresponding lower resolved voxel. Note that we only provide the networks with transport information on the low-resolution images, which might not be fully representative for the high-resolution image for the above-mentioned reason. Nevertheless, the simulations can be interpreted as a voxel-wise weighting based on their contribution to the overall flow, thereby guiding the networks towards important bottlenecks or connections within the transporting phase. Although this is not a replacement for the high-resolution information of the ground-truth images, we believe that it improves training efficiency and accuracy in effectively reconstructing the percolation paths.

The transport phenomena considered here are relevant to the physical processes occurring in battery electrodes, which involve two primary types of diffusive transport: ion transport through the electrolyte and electron transport through the conductive additives. The former is modeled by Fick’s law of diffusion, while electron transport is governed by the stationary potential equation, where the electron flow is driven by an electric field, which is proportional to the gradient of the electrical potential. Both transport processes can be mathematically described by the Laplace equation [34, 35]. Thus, the transport simulations are performed by solving the Laplace equation on voxel-based domains extracted from 3D image data of the material. The computational domain is then defined as follows. Given a binary image  $I: W \rightarrow \{0, 1\}$ , we first construct the continuous domains  $\Omega_{\text{ref},1}, \Omega_{\text{ref},0} \subset \mathbb{R}^3$  from the voxelized image representation, where  $\Omega_{\text{ref},1}$  corresponds to the solid phase and  $\Omega_{\text{ref},0}$  to the pore phase. For this, let the position vector  $\mathbf{p}_{(j,k,l)} \in \mathbb{R}^3$  associated with the voxel at  $(j, k, l) \in W$  be defined as  $\mathbf{p}_{(j,k,l)} = (j - 1, k - 1, l - 1) \in \mathbb{R}^3$ . The corresponding (embedded) voxel is then denoted by  $V_{(j,k,l)} = \mathbf{p}_{(j,k,l)} + [0, 1]^3$ . The domain corresponding to a certain phase is given by the union of voxels assigned to that phase. In particular, for the the domain  $\Omega_{\text{ref},1}$  of the solid phase, we have

$$\Omega_{\text{ref},1} = \bigcup_{(j,k,l) \in I^{-1}(1)} V_{(j,k,l)}, \quad (4)$$

and for the domain  $\Omega_{\text{ref},0}$  of the pore phase, it holds that

$$\Omega_{\text{ref},0} = \bigcup_{(j,k,l) \in I^{-1}(0)} V_{(j,k,l)}, \quad (5)$$

where  $I^{-1}(1)$  and  $I^{-1}(0)$  denote the sets of voxels associated with the solid and pore phase, respectively.

The transport simulations are conducted separately for the solid phase, on  $\Omega_{\text{ref},1}$ , and for the pore phase, on  $\Omega_{\text{ref},0}$ . This is done by numerically solving the Laplace equation in combination with certain boundary conditions by means of a finite element method [26, 27]. More precisely, for each phase

$i \in \{0, 1\}$ , the Laplace equation is given by

$$-\Delta u = 0 \quad \text{in } \mathring{\Omega}_{\text{ref},i}, \quad (6)$$

where  $\mathring{\Omega}_{\text{ref},i}$  denotes the interior of the domain  $\Omega_{\text{ref},i}$ . Dirichlet boundary conditions are prescribed on two opposing external surfaces of the cuboidal domain, setting the value of the function equal to 0 on one surface, and equal to 1 on the opposite surface. Zero-flux boundary conditions are imposed on all remaining surfaces, which include the remaining external faces as well as the internal interfaces between the solid and pore phases. This setup induces a gradient in the solution along the direction perpendicular to the Dirichlet boundary surfaces, which represents a concentration gradient in diffusion problems or a potential gradient in electrical conductivity problems. The solution in both material phases is depicted in Figure 2 in the last two columns.

After performing the above-described transport simulations for both the solid and the pore phase, we generate a discretized two-channel gray scale image  $\text{Diffu}(I): W \times \{0, 1\} \rightarrow (0, 1)$  that contains the solution for each respective phase. Let  $\mathbf{x}_c(j, k, l) = (j - 1/2, k - 1/2, l - 1/2)$  denote the center of the embedded voxel  $V_{(j,k,l)}$  for each  $(j, k, l) \in W$ . Let  $u_{\text{pore}}: \Omega_{\text{ref},0} \rightarrow (0, 1)$  denote the piecewise linear function resulting from solving the diffusion problem in the pore phase by means of a finite element method. Then, the first channel of the image is defined as

$$\text{Diffu}(I)(j, k, l, 0) = \begin{cases} u_{\text{pore}}(\mathbf{x}_c(j, k, l)), & \text{if } \mathbf{x}_c(j, k, l) \in \Omega_{\text{ref},0}, \\ 0, & \text{otherwise.} \end{cases} \quad (7)$$

Similarly, if  $u_{\text{solid}}: \Omega_{\text{ref},1} \rightarrow (0, 1)$  represents the solution in the solid phase, the second channel is defined as

$$\text{Diffu}(I)(j, k, l, 1) = \begin{cases} u_{\text{solid}}(\mathbf{x}_c(j, k, l)), & \text{if } \mathbf{x}_c(j, k, l) \in \Omega_{\text{ref},1}, \\ 0, & \text{otherwise.} \end{cases} \quad (8)$$

The resulting two-channel image encodes the transport characteristics of the material, capturing the spatial voxel-level solution of the transport problem across both phases, see Figure 3. This enriched dataset is used later on to guide the CNNs in accurately identifying critical transport paths, leading to an improved reconstruction of the nanostructure during super-resolution.

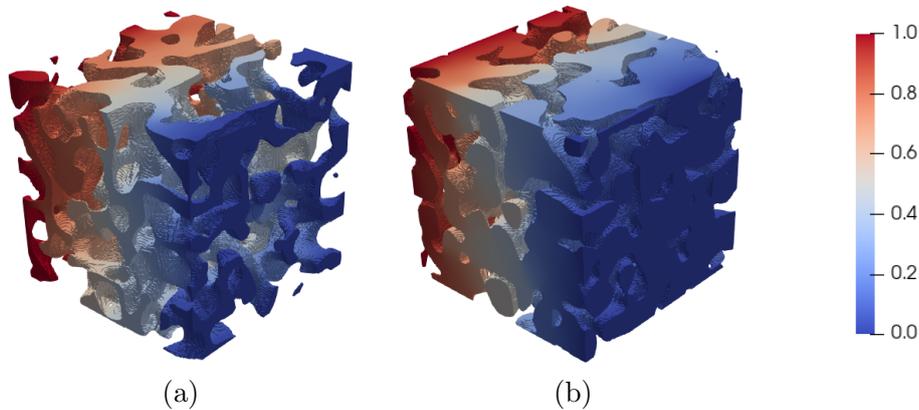


FIGURE 3. Solution to the diffusion problem within the reference volumes  $\Omega_{\text{ref},0}$  of pore phase (a) and  $\Omega_{\text{ref},1}$  of solid phase (b). The renderings in (a) and (b) correspond to a 3D view of the two right-most columns in Figure 2.

**2.1.5. Effective tortuosity.** Super-resolution techniques increase the resolution of 3D nanostructure data considered in the present paper. This enables a more precise representation of the underlying physical structures, which in turn improves the accuracy of transport simulations. These simulations can be either spatially resolved microscopic processes or macroscopic simulations in which homogenization techniques are used to average the properties across scales. When evaluating super-resolution in transport simulations, we focus on the effective tortuosity as a key metric. This metric is used to quantify how the nanostructure affects transport properties such as ion diffusion or electrical conductivity. In

the context of transport processes such as diffusion or conduction, tortuosity reflects how the structure of the material impedes transport and increases the effective resistance to diffusion or conduction compared to that of an idealized, homogeneous medium. A high tortuosity indicates convoluted paths and greater resistance to transport, while a low tortuosity indicates more direct paths. Effective tortuosity is particularly important when modeling transport behavior on larger scales, where the detailed nanostructure is averaged by homogenization processes.

In the literature [36–38], the effective tortuosity is commonly expressed as

$$\tau_{\text{eff}} = \varepsilon \frac{D}{D_{\text{eff}}}, \quad (9)$$

where  $\varepsilon$  is the volume fraction of the material phase in question. Here,  $D$  denotes the intrinsic transport coefficient, such as the diffusion or conductivity coefficient, depending on the specific transport process, while  $D_{\text{eff}}$  refers to the effective transport coefficient, which captures the averaged transport behavior within the heterogeneous microstructure. It accounts for the influence of the microstructure on the overall transport rate and is computed from transport simulations based on Eq. (6) and the boundary conditions described above. The value of  $D_{\text{eff}}$  has been computed following [12]. The ratio  $D/D_{\text{eff}}$  reflects the increase in path length or transport resistance imposed by the microstructural complexity of the material relative to a homogeneous medium.

By using super-resolution methods, finer details of the nanostructure that affect the transport paths, such as bottlenecks or percolation paths, can be captured more accurately. This leads to a more precise calculation of  $\tau_{\text{eff}}$  and enables better predictions in both microscopic and macroscopic simulations. Hence, improved resolution is thus crucial for accurately determining the relationship between the structure of materials and their effective transport properties.

**2.2. Neural networks for super-resolution.** In this section we describe the network architecture used for upsampling the virtual FIB/SEM image data of nanostructured active material in lithium-ion batteries. For super-resolving a low-resolution 3D input image, we consider three different up-scaling factors  $\alpha \in \{2, 4, 8\}$ . In addition, we consider two different types of input scenarios. In both cases, the input image is of the form

$$I_{\text{input}}: W \times \{1, \dots, c\} \rightarrow [0, 1], \quad (10)$$

where the number of channels  $c$  is either equal to 1 or to 3. The first case is the standard setting in which  $c = 1$  and the network only receives a low-resolution binary image  $I_{\text{LR}}: W \rightarrow \{0, 1\}$  to super-resolve. We denote this neural network by  $N_{\theta}^{\text{bin}, \alpha}$  depending on the scale factor  $\alpha$ , where  $\theta \in \mathbb{R}^m$  is the weight vector for some  $m \in \mathbb{N}$ . More precisely, in the following we consider  $N_{\theta}^{\text{bin}, \alpha}$  as a mapping which receives a single-channel 3D image  $I_{\text{input}}$  as input, defined as in Eq. (10) with  $c = 1$ , where  $I_{\text{input}}$  coincides with the low-resolution image  $I_{\text{LR}}$ , *i.e.*,

$$I_{\text{input}}(j, k, l, 1) = I_{\text{LR}}(j, k, l), \quad (11)$$

for each  $(j, k, l) \in W$ . Then, the output  $N_{\theta}^{\text{bin}, \alpha}(I_{\text{input}})$  of the neural network  $N_{\theta}^{\text{bin}, \alpha}$  is a super-resolved image

$$N_{\theta}^{\text{bin}, \alpha}(I_{\text{input}}): W_{\alpha} \rightarrow [0, 1], \quad (12)$$

where  $W_{\alpha}$  is defined as in Eq. (2). In the second case, we consider a network  $N_{\theta}^{\text{diffu}, \alpha}$  for any  $\alpha \in \{2, 4, 8\}$  and  $\theta \in \mathbb{R}^m$ , which additionally receives two further channels as input, namely, the solutions to the diffusion problem on solid and pore phase of the low-resolution image  $I_{\text{LR}}$ , as described in Section 2.1.4. More precisely, the input image  $I_{\text{input}}$  of the network  $N_{\theta}^{\text{diffu}, \alpha}$ , defined as in Eq. (10) with  $c = 3$ , is given by

$$I_{\text{input}}(j, k, l, c) = \begin{cases} I_{\text{LR}}(j, k, l), & \text{if } c = 1, \\ \text{Diffu}(I_{\text{LR}})(j, k, l, 0), & \text{if } c = 2, \\ \text{Diffu}(I_{\text{LR}})(j, k, l, 1), & \text{if } c = 3, \end{cases} \quad (13)$$

for each  $(j, k, l, c) \in W \times \{1, 2, 3\}$ , where the image  $\text{Diffu}(I_{\text{LR}})$  is defined as in Eqs. (7) and (8). Then, the network  $N_{\theta}^{\text{diffu}, \alpha}$  returns the high-resolution image  $N_{\theta}^{\text{diffu}, \alpha}(I_{\text{input}}): W_{\alpha} \rightarrow [0, 1]$ . In places where it is not relevant, we will omit the weight vector  $\theta$  and only write  $N^{\text{bin}, \alpha}$  and  $N^{\text{diffu}, \alpha}$ , respectively.

In both scenarios the first channel of the input image is always binary, while the output of the neural network is a gray scale image that can be interpreted as a fuzzy classification map, where a value closer to 0 or 1 indicates a larger certainty of membership towards the pore or solid phase, respectively. A thresholding will be applied to the output images of the neural networks after training to ensure that the final results are also binary. Details on the determination of the thresholding parameter are given at the end of Section 2.2.3.

**2.2.1. Network architecture.** The architecture of the neural networks considered in the present paper is a slight modification of SRResNet [39], which has already been used as a generator in a generative adversarial network (GAN) for super-resolving microscopic image data [40]. More precisely, the networks  $N^{\text{bin},\alpha}$ ,  $\alpha \in \{2, 4, 8\}$ , introduced above deploy a total of 8 residual blocks [41], which consist of convolutional layers followed by an element-wise addition of the output of the convolutional layer and the input to the residual block. This type of element-wise addition is called a skip connection, which is used to counteract the problem of vanishing gradients during training. Upsampling is achieved using a PixelShuffle3D layer [42] followed by a ReLU layer [43]. Depending on the desired scaling factor  $\alpha \in \{2, 4, 8\}$  of the super resolution, pairs of PixelShuffle3D and ReLU layers are deployed one, two, or three consecutive times before the output layer. A schematic overview of the network architecture is shown in Figure 4.

For each  $\alpha \in \{2, 4, 8\}$ , the first layer of the network  $N^{\text{diffu},\alpha}$  differs slightly compared to  $N^{\text{bin},\alpha}$ . In this scenario, the input is a three-channel image in which the first channel is the binary image, and the remaining two channels are given by the solution to the diffusion problem in the solid and in the pore phase given in Eq. (13), see Section 2.1.4. In this case, the size of the first convolutional layer needs to be adapted. The remaining network architecture is unchanged.

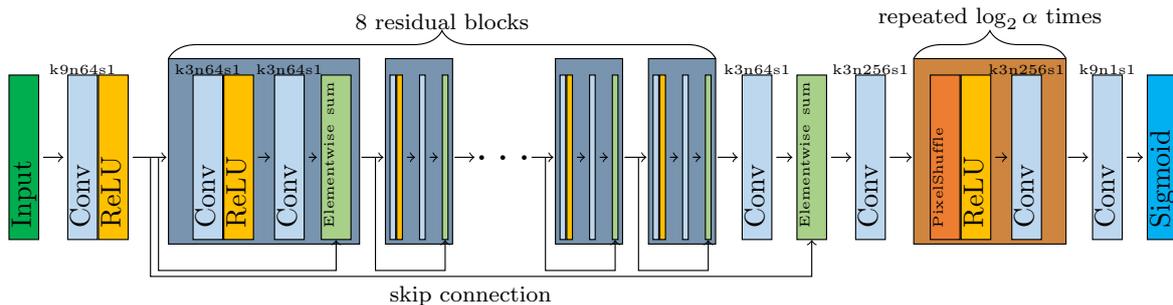


FIGURE 4. Schematic overview of network architecture.

**2.2.2. Training, validation and testing sets.** The artificially generated binary images presented in Section 2.1.3 and their corresponding two-channel gray scale images containing the solutions to the diffusion problem on solid and pore phase presented in Section 2.1.4 constitute our database of simulated image data. More precisely, pairs of low-resolution and high-resolution images for training, validation, and testing super-resolution with a scaling factor of  $\alpha \in \{2, 4, 8\}$  are comprised as follows. For any given  $s \in \{1, \dots, 5\}$  and  $i \in \{1, \dots, 10\}$ , we set the target (or ground truth) image  $I_{\text{HR}}$  to the realization  $I_{800}^{s,i}$  of the stochastic 3D model, as described in Section 2.1.2; the corresponding low-resolution image is given by  $I_{\text{LR}} = I_{800/\alpha}^{s,i}$ , see Eq. (3) for  $\alpha = 2$ . Then, the input image  $I_{\text{input}}$  for the networks  $N^{\text{bin},\alpha}$  and  $N^{\text{diffu},\alpha}$ ,  $\alpha \in \{2, 4, 8\}$ , is given by means of Eqs. (11) and (13), respectively. In this manner, for each network to be trained (*i.e.*, two networks  $N^{\text{bin},\alpha}$  and  $N^{\text{diffu},\alpha}$  are considered for each  $\alpha \in \{2, 4, 8\}$ ), we can derive a pair  $(I_{\text{input}}, I_{\text{HR}})$  of input and target images for any  $s \in \{1, \dots, 5\}$  and  $i \in \{1, \dots, 10\}$ .

We split this data into training, validation, and testing sets as follows: For each  $s \in \{1, \dots, 5\}$ , the model realizations  $I_{800}^{s,i}$ ,  $i \in \{1, \dots, 7\}$ , are used to determine pairs  $(I_{\text{input}}, I_{\text{HR}})$  of training data, resulting in  $7 \times 5 = 35$  pairs. Similarly, the realizations  $I_{800}^{s,i}$ , for  $s \in \{1, \dots, 5\}$  and  $i \in \{8, 9\}$ , are used for the validation set, *i.e.*, the corresponding pairs  $(I_{\text{input}}, I_{\text{HR}})$  will be used to validate the network performance during training and to define an early-stopping criterion. More information on this will be given in Section 2.2.3. Finally, the remaining realizations  $I_{800}^{s,i}$ , for  $s \in \{1, \dots, 5\}$  and  $i = 10$ , are used to

define the testing set, *i.e.*, the corresponding pairs of images will be used to test the performance of the networks after training using the descriptors presented in Section 2.3. Furthermore, we will assess the quality of the super-resolution results not only with respect to the testing set comprised of simulated image data, but also with respect to the experimentally measured tomographic image data described in Section 2.1.1. The results of this analysis of network performance on simulated and tomographic image will be presented separately in Section 3.

**2.2.3. Network training.** For the optimization of the weights  $\theta \in \mathbb{R}^m$  of the neural networks, we train the networks based on pairs of virtual low- and high-resolution images, as described in Section 2.2.2. The loss function that we consider is the so-called binary cross-entropy loss function  $\lambda: \{0, 1\} \times (0, 1) \rightarrow [0, \infty)$ , given by

$$\lambda(y, \tilde{y}) = -y \log \tilde{y} - (1 - y) \log(1 - \tilde{y}), \quad (14)$$

for any  $y \in \{0, 1\}$  and  $\tilde{y} \in (0, 1)$ . This loss function will be applied voxel-wise to the image predicted by the neural networks and the corresponding target image of the training data. Note that the second argument of the binary cross-entropy loss function  $\lambda$  is a real number of the interval  $(0, 1)$ , as these are the possible values of images predicted by the neural networks considered in the present paper. They can be interpreted as the probability that a voxel belongs to the solid phase. Then, in order to quantify the discrepancy between some binary image  $I$  and a  $(0,1)$ -valued image  $\tilde{I}$  defined over the same domain  $W$ , as given in Eq. (1), we use the loss function  $\Lambda$  given by

$$\Lambda(I, \tilde{I}) = \sum_{w \in W} \lambda(I(j, k, l), \tilde{I}(j, k, l)), \quad (15)$$

for any  $I: W \rightarrow \{0, 1\}$  and  $\tilde{I}: W \rightarrow (0, 1)$ . Then, the corresponding optimization problem for training a neural network denoted by  $N_\theta$  (*i.e.*,  $N_\theta^{\text{bin}, \alpha}$  or  $N_\theta^{\text{diffu}, \alpha}$ ,  $\alpha \in \{2, 4, 8\}$ ) is given by

$$\theta^* = \operatorname{argmin}_{\theta \in \Theta} \mathbb{E} \left( \Lambda(N_\theta(\mathbf{I}_{\text{input}}), \mathbf{I}_{\text{HR}}) \right), \quad (16)$$

where  $\mathbf{I}_{\text{input}}$  and  $\mathbf{I}_{\text{HR}}$  are randomly chosen matching sub-images of the training data, *i.e.*, the random images  $\mathbf{I}_{\text{input}}$  and  $\mathbf{I}_{\text{HR}}$  depict the same morphology at different resolutions. The expectation in the optimization problem given in Eq. (16) is taken over the random images  $\mathbf{I}_{\text{input}}$  and  $\mathbf{I}_{\text{HR}}$ .

The optimization algorithm used to solve the minimization problem given in Eq. (16) is Adam [44]. Training is performed on random  $24 \times 24 \times 24$  cutouts of the low-resolution training data, *i.e.*, in Eq. (1) we set  $m_1 = m_2 = m_3 = 24$ . More precisely, during training, we first select a random training pair  $(I_{\text{input}}, I_{\text{HR}})$  from the training data defined in Section 2.2.2. Then, a random  $24 \times 24 \times 24$  cutout is chosen from the (single- or three-channel) image  $I_{\text{input}}$ . The corresponding  $\alpha 24 \times \alpha 24 \times \alpha 24$  cutout of the image  $I_{\text{HR}}$  is used as the target image. Subsequently, data augmentation is performed by mirroring the cutouts with a 50% chance and applying a random rotation from the symmetry group of the cube, *e.g.*, rotations around the coordinate axes by multiples of  $\pi/2$ . For the latter, note that the chosen rotation can also be the identity. In the scenario where the CNN also receives information from numerical simulations, we apply the chosen transformation to all image channels. After augmentation, we obtain a random pair  $(\mathbf{I}_{\text{input}}, \mathbf{I}_{\text{HR}})$  of images. One training step consists of generating  $b > 0$  (referred to as batch size) such pairs, which are used to compute an estimate of the expectation given in Eq. (16) by means of Monte-Carlo simulation. Based on this estimate of expected loss, the weights  $\theta$  of the network are adapted using the Adam optimization method, which concludes an individual training step.

Overall, the training procedure consists of a maximum of 1000 epochs, in each of which 400 training steps are performed. After every epoch, an early stopping check is performed by estimating the expected loss by generating pairs of random images from the validation data set instead of the training data set, see Section 2.2.2. For this, 30 random cutouts of size  $24 \times 24 \times 24$  are chosen to compute an estimate for the expected loss by Monte-Carlo simulation. If the estimate of the expected loss does not improve during 10 consecutive epochs, the training is stopped. The batch size  $b$  of each step is equal to 8 for the cases  $\alpha \in \{2, 4\}$  and equal to 1 in the case  $\alpha = 8$  due to memory limitations. The networks corresponding to different scaling factors  $\alpha \in \{2, 4, 8\}$  share most of their network architecture and therefore have nearly identically structured weight vectors, enabling us to use transfer learning. For example, the weights  $\theta$  of the network  $N_\theta^{\text{bin}, \alpha}$  are partially initialized—for coinciding layers—with the

weights of the trained network with scaling factor  $\alpha/2$  for  $\alpha \in \{4, 8\}$ . Similarly, transfer learning is deployed for training  $N_\theta^{\text{diffu},\alpha}$  with  $\alpha \in \{4, 8\}$ .

Recall that the output of the neural networks is a gray scale image with values in  $(0, 1)$ , which can be interpreted as the probability of the respective voxel belonging to the solid phase. This image needs to be thresholded in order to obtain a proper binary high-resolution version of the binary low-resolution input. Let  $I^{\geq\mu}: W \rightarrow \{0, 1\}$  denote the binary image resulting from the thresholding of a gray scale image  $I: W \rightarrow [0, 1]$  at the level  $\mu \in (0, 1)$ , given by

$$I^{\geq\mu}(x) = \begin{cases} 1, & \text{if } I(x) \geq \mu, \\ 0, & \text{otherwise.} \end{cases} \quad (17)$$

The optimal thresholding parameter  $\mu^* \in (0, 1)$  to be deployed on network outputs is determined by minimizing the average error in porosity. If  $\varepsilon(I) \in [0, 1]$  is the porosity of a binary image  $I: W \rightarrow \{0, 1\}$ , *i.e.*, the fraction of voxels belonging to the pore phase, then we define  $\mu^* \in (0, 1)$  by

$$\mu^* = \operatorname{argmin}_{\mu \in (0,1)} \mathbb{E} |\varepsilon(N(\mathbf{I}_{\text{input}})^{\geq\mu}) - \varepsilon(\mathbf{I}_{\text{HR}})|. \quad (18)$$

In order to solve this minimization problem, all possible  $24 \times 24 \times 24$  cutouts of low-resolution images in the validation set  $\mathcal{E}^\alpha$  are used. The optimal thresholding parameter is then found by use of the Nelder-Mead method [45].

Note that the minimization problem given in Eq. (18) is solved for each neural network that has been trained, *i.e.*, for each  $N^{\text{bin},\alpha}$  and  $N^{\text{diffu},\alpha}$  with  $\alpha \in \{2, 4, 8\}$ . The corresponding thresholds  $\mu^*$  that have been identified for the individual trained network architectures are denoted by  $\mu^{\text{bin},\alpha}$  and  $\mu^{\text{diffu},\alpha}$  for  $\alpha \in \{2, 4, 8\}$ , respectively.

**2.3. Metrics for validation.** In this section, we describe the metrics that we use to assess the quality of the super-resolution results achieved by the neural networks  $N^{\text{bin},\alpha}$  and  $N^{\text{diffu},\alpha}$ , with  $\alpha \in \{2, 4, 8\}$ . For the definition of the metrics, we assume that we are given a pair  $(I_{\text{input}}, I_{\text{HR}})$  of images consisting of the low-resolution input  $I_{\text{input}}$  and the high-resolution ground truth image  $I^{\text{HR}}: W \rightarrow \{0, 1\}$  defined over some rectangular cuboid  $W \subset \mathbb{Z}^3$ . Let  $I^{\text{SR}}$  denote the super-resolved image given by either  $I^{\text{SR}} = N^{\text{bin},\alpha}(I_{\text{input}})^{\geq\mu^{\text{bin},\alpha}}$  or  $I^{\text{SR}} = N^{\text{diffu},\alpha}(I_{\text{input}})^{\geq\mu^{\text{diffu},\alpha}}$ , depending on the considered scenario.

**2.3.1. Accuracy.** The accuracy is a straightforward validation of the training procedure. We just determine the relative number of correctly assigned voxels in the  $I^{\text{SR}}$ , that is

$$\operatorname{acc}(I^{\text{SR}}, I^{\text{HR}}) = \frac{1}{|W|} \sum_{x \in W} \mathbb{1}_{\{x' \in W: I^{\text{SR}}(x') = I^{\text{HR}}(x')\}}(x), \quad (19)$$

where  $|W|$  denotes the cardinality of  $W$ , and  $\mathbb{1}_A$  is the indicator over some set  $A$ , *i.e.*,  $\mathbb{1}_A(x) = 1$  if and only if  $x \in A$ ; otherwise  $\mathbb{1}_A(x) = 0$  holds. Note that this concept of accuracy is closely linked to the metric used for network training in the optimization problem (16), as the function  $\Lambda$  defined in Eq. (15) also performs a voxel-wise comparison between artificial super-resolution and ground truth.

**2.3.2. Surface accuracy.** The task of correctly classifying a voxel in a super-resolved image is typically easier if the voxel belongs to a larger area contained within the same phase. On the other hand, voxels belonging to boundary regions where two phases meet are harder to classify correctly because it is not clear where precisely the phase boundary runs in the high-resolution image. For this reason, we consider a modified notion of the voxel-wise accuracy defined above, which only counts correctly classified voxels that are near the boundary of the solid and pore phase. In order to determine the boundary region of the high-resolution image  $I^{\text{HR}}$ , we make use of morphological erosion [46], which is defined as follows. Given a so-called structuring element  $S \subset \mathbb{R}^3$ , the erosion  $I^{\text{HR}} \ominus S$  is defined by

$$(I^{\text{HR}} \ominus S)(x) = \min_{y \in S \cap \mathbb{Z}^3} I^{\text{HR}}(x + y), \quad (20)$$

for each  $x \in W$ , where we put  $I^{\text{HR}}(z) = 1$  for all  $z \in \mathbb{Z}^3 \setminus W$ . For our application, we choose  $S$  to be a ball centered at the origin with a radius of  $1.1\alpha/2.0 = 0.55\alpha$  voxels, where  $\alpha \in \{2, 4, 8\}$  is the scaling factor of the super-resolution task. The image  $I^{\text{HR}} - (I^{\text{HR}} \ominus S)$  labels the voxels within the solid phase

with a distance of less than  $0.55\alpha$  voxels to the boundary. Furthermore, we apply the same process to the pore phase and combine the results. The binary image  $\partial I^{\text{HR}}$  given by

$$\partial I^{\text{HR}} = I^{\text{HR}} - (I^{\text{HR}} \ominus S) - (1 - I^{\text{HR}}) \left( (1 - I^{\text{HR}}) \ominus S \right), \quad (21)$$

satisfies  $\partial I^{\text{HR}}(x) = 1$  if and only if  $x \in W$  is a voxel with a distance of less than  $1.1\alpha$  voxels to the boundary of the pore and solid phase. Thus, the set  $B = (\partial I^{\text{HR}})^{-1}(1) \subset W$  contains precisely those voxels that belong to the boundary region of the image  $I^{\text{HR}}$ . We can now define the surface accuracy by

$$\text{surfacc}(I^{\text{SR}}, I^{\text{HR}}) = \frac{1}{|B|} \sum_{x \in B} \mathbb{1}_{\{x' \in W: I^{\text{SR}}(x') = I^{\text{HR}}(x')\}}(x). \quad (22)$$

**2.3.3. Relative error of mean geodesic tortuosity.** In order to assess the quality of the super-resolved images not only with respect to voxel-wise comparisons, but also with respect to more global properties of the nanostructure, we consider the mean geodesic tortuosity  $\tau_{\text{geod}}$  which is a geometric descriptor that quantifies the length of transport paths through the material within a given phase [22]. In the present paper, we always consider paths that traverse the sampling window along an arbitrarily chosen direction. Under the assumption of isotropy, the choice of this direction does not influence the resulting error statistic. At first, the geodesic tortuosity is defined for each voxel within the starting plane that belongs to a given phase as the length of a shortest path traversing the material through the chosen phase, divided by the thickness of the material. The mean geodesic tortuosity is then computed by averaging over all values of geodesic tortuosity for all voxels in the starting plane that belong to the chosen phase. For a formal description of geodesic tortuosity in the context of random closed sets, we refer to [47]. Now let  $\tau_{\text{geod}}(I^{\text{SR}})$  and  $\tau_{\text{geod}}(I^{\text{HR}})$  denote the values of mean geodesic tortuosity of the super-resolved image and the high-resolution ground truth image, respectively. Then the accuracy of the super-resolved image with respect to mean geodesic tortuosity is given by the relative error

$$\tau_{\text{geod}}^{\text{acc}}(I^{\text{SR}}, I^{\text{HR}}) = 1 - \frac{|\tau_{\text{geod}}(I^{\text{SR}}) - \tau_{\text{geod}}(I^{\text{HR}})|}{\tau_{\text{geod}}(I^{\text{HR}})}. \quad (23)$$

**2.3.4. Relative error of effective tortuosity.** In addition to the purely geometric notion of geodesic tortuosity, we also consider the effective tortuosity. As stated in Section 2.1.5, the effective tortuosity  $\tau_{\text{eff}}$  quantifies the influence of the microstructure on the overall transport rate based on the Laplace equation given in Eq. (6). Unlike geodesic tortuosity, which only considers shortest paths through the nanostructure for any given starting point, effective tortuosity considers the influence of nanostructure globally through transport simulations on the entire domain  $\Omega_{\text{ref}}$ . Therefore, aspects such as bottleneck effects that restrict flow through tight spaces are also quantified by effective tortuosity, whereas geodesic tortuosity neglects them. Analogously to the case of mean geodesic tortuosity, we define the accuracy of the super-resolved image with respect to effective tortuosity by the relative error

$$\tau_{\text{eff}}^{\text{acc}}(I^{\text{SR}}, I^{\text{HR}}) = 1 - \frac{|\tau_{\text{eff}}(I^{\text{SR}}) - \tau_{\text{eff}}(I^{\text{HR}})|}{\tau_{\text{eff}}(I^{\text{HR}})}. \quad (24)$$

### 3. RESULTS AND DISCUSSION

We evaluate the effectiveness of the trained neural networks in performing super-resolution with respect to the validation metrics stated in Section 2.3. On the one hand, we carry out this validation on the test set consisting of simulated image data that was not used for training as explained in Section 2.2.2. On the other hand, we also consider the final application of interest, which is to super-resolve experimentally measured tomographic image data. Furthermore, as a baseline reference for the neural networks considered in this study, we determine super-resolved images by means of classical tri-quadratic interpolation [21] in each case and compute the validation metrics for these images.

Figure 5 shows a visual comparison between the super-resolved image data for a scaling factor of  $\alpha = 8$  and the ground truth, where the voxels displayed in blue are incorrectly classified as pore phase, while the voxels displayed in magenta are incorrectly classified as solid phase. We can clearly observe that the image data super-resolved by means of a neural network heavily outperforms the super-resolution by interpolation with respect to incorrectly classified voxels. However, the difference in

performance between networks  $N^{\text{bin},\alpha}$  and  $N^{\text{diffu},\alpha}$  is not so clear. Further 2D slices of super-resolved tomographic image data are provided in Figure A1 of the appendix.

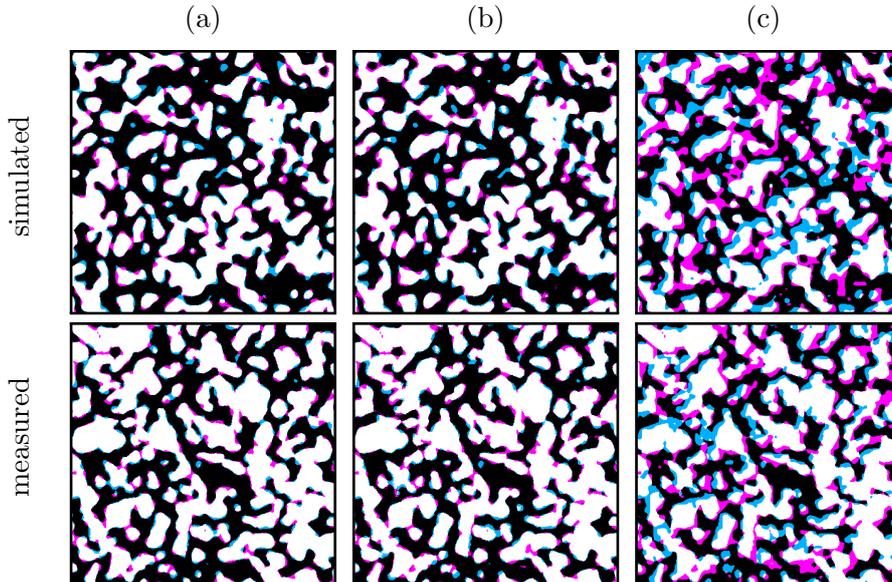


FIGURE 5. Planar sections through 3D images visualizing the voxel-wise error in artificially super-resolved image data in comparison to ground truth. Top row: Simulated image data. Bottom row: Experimentally measured image data. Voxels that are correctly classified as solid or pore voxels are displayed in white and black, respectively, while voxels that are incorrectly classified as solid and pore voxels are displayed in magenta and blue, respectively. Super-resolution by a scaling of  $\alpha = 8$  has been performed by (a)  $N^{\text{bin},8}$ , (b)  $N^{\text{diffu},8}$  and (c) tri-quadratic interpolation.

We now turn to the quantitative comparison of the quality of the super-resolution results, using the metrics stated in Section 2.3, see the box plots shown in Figure 6. For experimentally measured image data, the 3D images of each of the 5 measured samples are first downsampled, analogously to the procedure stated in Eq. (3), to obtain ground truth images at lower resolutions. Subsequently, the resulting images are subdivided into 8 equally sized, non-overlapping images. Starting from the lowest resolution, super-resolved images with scaling factors  $\alpha \in \{2, 4, 8\}$  were computed. For simulated image data, the same subdivision was applied to the realizations of the stochastic model  $I_{800/\alpha}^{s,10}$ , for each  $s \in \{1, \dots, 5\}$  and  $\alpha \in \{1, 2, 4, 8\}$ . The validation metrics stated in Section 2.3 are then evaluated on the super-resolved sub-images for each scaling factor  $\alpha \in \{2, 4, 8\}$ . Note that these images have not been used during the training process of the neural networks. This results in a total of  $8 \times 5 = 40$  data points entering each individual box in Figure 6.

Figure 6a depicts the results obtained for the validation metric  $\text{acc}$  given in Eq. (19), which is essentially a quantification of the differences shown in Figure 5. We observe that the interpolation approach performs much worse than super-resolution by use of the neural networks. This result is obviously expected, as interpolation is a general approach that does not require any training data. However, the increase in performance when these additional steps are taken is impressively significant. Furthermore, as the scaling factor increases, the accuracy of the interpolation decreases. This might be expected, as the difficulty of the super-resolution problem naturally increases with a higher scaling factor. On the other hand, both neural network approaches show a much more stable performance across the scenarios  $\alpha \in \{2, 4, 8\}$ . In Figure 6b, we see that, as expected, the values for the surface accuracy are lower compared to values for the accuracy in all scenarios. However, the qualitative relationship between the considered cases is similar to that of Figure 6a. With respect to accuracy and surface accuracy, the performance of the neural networks  $N^{\text{bin},\alpha}$  and  $N^{\text{diffu},\alpha}$  is fairly similar.

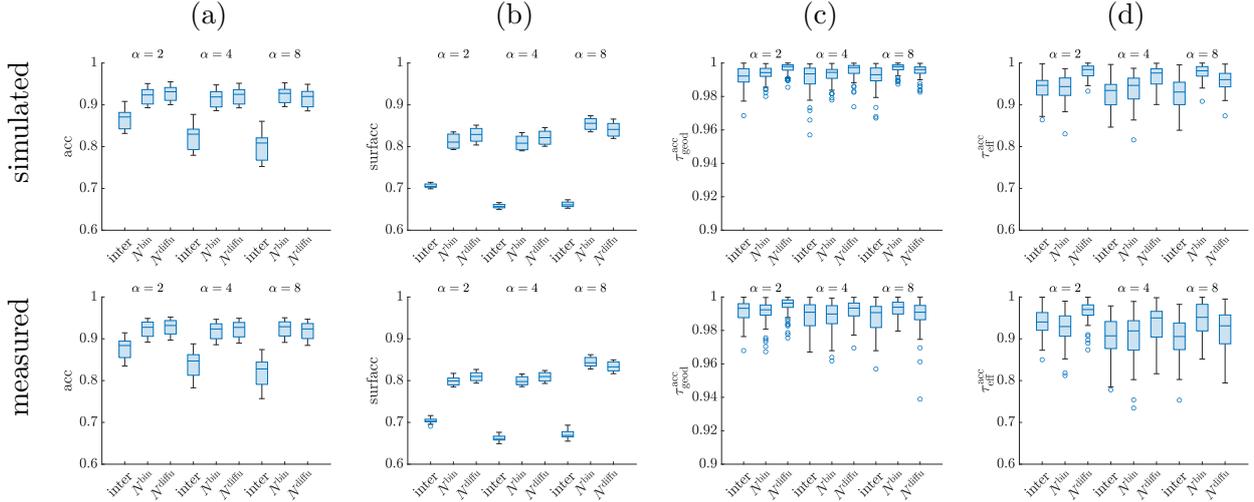


FIGURE 6. Validation metrics of super-resolutions for each scaling factor  $\alpha \in \{2, 4, 8\}$ . Top row: Simulated image data. Bottom row: Tomographic image data. Shown are the results obtained for accuracy (a), surface accuracy (b), and the relative error of mean geodesic (c) and effective (d) tortuosity. Box plots display the mean value as well as lower and upper quartiles. The whiskers extending from the boxes show the minimum and maximum values. Outliers are detected using inter-quartile range and marked by a circle.

It should be noted that the performance of the interpolation with respect to the voxel-wise metrics shown in Figures 6a and 6b significantly decreases with an increasing scaling factor  $\alpha$ , while the performance of the two neural networks remains almost constant for different values of  $\alpha$ . Furthermore, recall that the values of mean geodesic and effective tortuosity depend on the considered phase. The box plots shown in Figures 6c and 6d depict the relative errors of these tortuosities with respect to the ground truth, averaged over all analyzed cutouts for both the pore phase and the solid phase. A visualization showing the relative errors separately for the pore and solid phase can be found in Figure A2 of the Appendix.

When comparing the results obtained by the two neural networks, we see that  $N^{\text{diffu},\alpha}$  outperforms  $N^{\text{bin},\alpha}$  at the scaling factors  $\alpha \in \{2, 4\}$ . However, a slightly worse performance is observed at the highest scaling factor  $\alpha = 8$ . This is consistent across all validation metrics considered in Figure 6. In order to investigate this contrary behavior for  $\alpha = 8$  more closely, we have retrained the involved neural networks to check whether this is caused by the random initialization of trainable weights before the training process. The metrics have also been evaluated on the training and testing data in order to investigate whether possible over-fitting phenomena are present. If that were the case, we would have observed different behavior in the accuracies for the super-resolved training data. However, the comparison between the two networks remained almost identical in all the cases considered. Therefore, we believe that neither random weight initialization nor a possible occurrence of over-fitting has caused the discrepancy at scale  $\alpha = 8$ , so that the reason for this behavior ultimately remains unclear.

The neural networks considered in this study have only been trained on artificial image data generated by means of the stochastic 3D nanostructure model described in Section 2.1.2. There are some differences between model realizations and tomographic image data that can be visually detected by the eye, see Figure 2. In particular, as hierarchically structured NMC111 particles are an agglomerate of many smaller primary particles, one can still observe rough edges of the original primary particles' shapes. These rough edges are not properly represented in realizations of the stochastic model, in which the interface between solid and pore phase forms a relatively smooth boundary surface. However, this effect is only a visual detail, as the quantitative validation performed in [12], based on geometrical descriptors and effective transport properties, showed a high agreement. Moreover, note that for all metrics considered in Figure 6, the validation with respect to tomographic image data leads to values which are nearly identical to those obtained for simulated image data. This shows that the predictive

power of the neural network applied to experimentally measured tomographic image data is not limited by the difference between simulated image data and measured tomographic image data, but rather by the complexity of the super-resolution problem itself.

#### 4. CONCLUSIONS

This study presents an innovative approach to generate high-resolution 3D image data of porous materials by integrating super-resolution techniques with spatially resolved transport simulations. By using a stochastic model to simulate virtual samples, we were able to efficiently generate digital twins of nanostructured materials that were used to train CNNs for super-resolution tasks. This method significantly reduces the dependence on large amounts of high-resolution experimental data, as virtual data sets serve as effective training data for neural networks.

The integration of transport simulations was found to be advantageous in supporting CNNs in the accurate identification of important transport pathways only for scaling factors  $\alpha \in \{2, 4\}$ , while for  $\alpha = 8$  the network without transport simulations performed better. It remains unclear whether CNNs trained with additional transport information can truly perform better in classifying boundary voxels and correctly representing material transport properties at high scaling factors. Additional investigations will be necessary to come to a clear conclusion. Our analysis is based on validation metrics that include a voxel-wise accuracy as well as two notions of tortuosity that quantify the length of transport paths through the material.

The results show that the networks trained on artificially generated data generalize well to measured tomographic data, confirming the robustness of this approach. The framework developed in this study is applicable not only to battery materials but also to other porous structures in various energy storage and conversion technologies. Furthermore, the virtual training data generated in this study paves the way for additional computer experiments, that can be explored in future work. In particular, one promising direction would be to investigate how insights from high-resolution numerical solutions during training can improve a neural network’s ability to super-resolve domain morphologies. More precisely, neural networks could be trained to receive low-resolution multi-channel images of domain geometry and associated numerical solutions as input and output corresponding high-resolution multi-channel images of both morphology and numerical solutions. Consequently, neural networks are provided with high-resolved numerical solutions during training. Once trained, their performance can be evaluated with respect to (i) their ability to super-resolve the morphology and (ii) their capability to reconstruct the corresponding numerical solution. In addition to this future investigation, one can explore how effectively super-resolving neural networks can learn to ‘solve’ the diffusion problem considered here, without the additional information of the low-resolution numerical solution. For this purpose, networks can be trained that receive only low-resolution images of the morphology as input, with the output being high-resolution numerical solutions. Such an approach could reduce both the time and resources required for high-resolution imaging and numerical simulations of diffusion problems.

#### DATA AVAILABILITY

The datasets generated during and/or analyzed during the current study are available from the corresponding authors on reasonable request.

#### CODE AVAILABILITY

All formulations and algorithms necessary to reproduce the results of this study are described in the Methods and Results sections.

#### ACKNOWLEDGMENTS

We gratefully acknowledge partial funding for this research paper from dtec.bw – the Digitalization and Technology Research Center of the Bundeswehr through the projects "CTCentre" and "Digital Materials Foundry". We also appreciate the provision of computational resources (HPC cluster HSU-per). Dtec.bw is supported by the European Union – NextGenerationEU. In addition, this research was supported by the German Federal Ministry for Economic Affairs and Climate Action (BMWK)

and granted through Project Management Jülich (03ETE039A, 03ETE039G, 03ETE039H). Furthermore, partial funding was provided by the German Research Foundation (DFG) within the framework of SPP 2248 “Polymer-based batteries” (CA 633/4-2, MA 5039/7-2, SCHM 997/39-2). The authors thank Leon Schröder for fruitful discussions on the training data used in this study.

#### CREDIT AUTHORSHIP CONTRIBUTION STATEMENT

**Orkun Furat:** Conceptualization, Data curation, Investigation, Methodology, Software, Validation, Writing–original draft, Writing–review & editing. **Phillip Gräfensteiner:** Conceptualization, Data curation, Formal analysis, Methodology, Software, Validation, Visualization, Writing–original draft, Writing–review & editing. **Rishabh Saxena:** Data curation, Investigation, Software, Visualization, Writing–original draft, Writing–review & editing. **Markus Osenberg:** Data curation, Investigation, Resources, Visualization, Writing–review & editing. **Matthias Neumann:** Methodology, Software, Writing–review & editing. **Ingo Manke:** Funding acquisition, Resources, Supervision, Writing–review & editing. **Thomas Carraro:** Conceptualization, Funding acquisition, Investigation, Software, Supervision, Project administration, Writing–original draft, Writing–review & editing. **Volker Schmidt:** Funding acquisition, Project administration, Supervision, Writing–review & editing.

#### COMPETING INTERESTS

The authors declare that they have no competing financial or non-financial interests.

#### REFERENCES

- [1] S. Torquato. *Random Heterogeneous Materials: Microstructure and Macroscopic Properties*. Springer, 2002.
- [2] Y. Zhang and W. Yu. Comparison of DEM super-resolution methods based on interpolation and neural networks. *Sensors*, 22:745, 2022.
- [3] R. Soltanmohammadi and S. A. Faroughi. A comparative analysis of super-resolution techniques for enhancing micro-CT images of carbonate rocks. *Applied Computing and Geosciences*, 20:100143, 2023.
- [4] S. N. Chiu, D. Stoyan, W. S. Kendall, and J. Mecke. *Stochastic Geometry and Its Applications*. J. Wiley & Sons, 2013.
- [5] D. Jeulin. *Morphological Models of Random Structures*. Springer, 2021.
- [6] S. Kim, H. Lee, J. Lim, J. Park, and Y. M. Lee. Digital twin battery modeling and simulations: A new analysis and design tool for rechargeable batteries. *ACS Energy Letters*, 9:5225–5239, 2024.
- [7] S. R. Kalidindi, M. Buzzy, B. L. Boyce, and R. Dingreville. Digital twins for materials. *Frontiers in Materials*, 9:818535, 2022.
- [8] K. Schladitz, C. Redenbach, T. Barisin, C. Jung, N. Jeziorski, L. Bosnar, J. Fulir, and P. Gospodnetić. Simulation of microstructures and machine learning. In F. Willot, J. Dirrenberger, S. Forest, D. Jeulin, and A. V. Cherkaev, editors, *Continuum Models and Discrete Systems*, pages 243–256. Springer, 2024.
- [9] C. Fend, A. Moghiseh, C. Redenbach, and K. Schladitz. Reconstruction of highly porous structures from FIB-SEM using a deep neural network trained on synthetic images. *Journal of Microscopy*, 281:16–27, 2021.
- [10] A. Tsamos, S. Evsevlev, R. Fioresi, F. Faglioni, and G. Bruno. Synthetic data generation for automatic segmentation of X-ray computed tomography reconstructions of complex microstructures. *Journal of Imaging*, 9:22, 2023.
- [11] T. Sardhara, R. C. Aydin, Y. Li, N. Piché, R. Gauvin, C. J. Cyron, and M. Ritter. Training deep neural networks to reconstruct nanoporous structures from FIB tomography images using synthetic training data. *Frontiers in Materials*, 9:837006, 2022.
- [12] M. Neumann, S. E. Wetterauer, M. Osenberg, A. Hilger, P. Gräfensteiner, A. Wagner, N. Bohn, J. R. Binder, I. Manke, T. Carraro, and V. Schmidt. A data-driven modeling approach to quantify morphology effects on transport properties in nanostructured NMC particles. *International Journal of Solids and Structures*, 280:112394, 2023.

- [13] S. Jang, Y. Kang, H. Kim, J. Park, and K. T. Lee. Digital twin reveals the impact of carbon binder domain distribution on performance of lithium-ion battery cathodes. *Small Structures*, 6:2400350, 2024.
- [14] J. Piruzjam, G. Liu, L. Rubacek, M. Frey, and T. Carraro. On the analytical solution of single particle models and semi-analytical solution of P2D model for lithium-ion batteries. *Electrochimica Acta*, 492:144259, 2024.
- [15] F. Brosa Planella, W. Ai, A. M. Boyce, A. Ghosh, I. Korotkin, S. Sahu, V. Sulzer, R. Timms, T. G. Tranter, M. Zyskin, S. J. Cooper, J. S. Edge, J. M. Foster, M. Marinescu, B. Wu, and G. Richardson. A continuum of physics-based lithium-ion battery models reviewed. *Progress in Energy*, 4:042003, 2022.
- [16] A. Schmidt, E. Ramani, T. Carraro, J. Joos, A. Weber, M. Kamlah, and E. Ivers-Tiffée. Understanding deviations between spatially resolved and homogenized cathode models of lithium-ion batteries. *Energy Technology*, 9:2000881, 2021.
- [17] V. Sulzer, S. G. Marquis, R. Timms, M. Robinson, and S. J. Chapman. Python battery mathematical modelling (PyBaMM). *Journal of Open Research Software*, 9:14, 2021.
- [18] H. Hamed, S. Yari, J. D’Haen, F. U. Renner, N. Reddy, A. Hardy, and M. Safari. Demystifying charge transport limitations in the porous electrodes of lithium-ion batteries. *Advanced Energy Materials*, 10:2002492, 2020.
- [19] S. Hein, T. Danner, D. Westhoff, B. Prifling, R. Scurtu, L. Kremer, A. Hoffmann, A. Hilger, M. Osenberg, I. Manke, M. Wohlfahrt-Mehrens, V. Schmidt, and A. Latz. Influence of conductive additives and binder on the impedance of lithium-ion battery electrodes: Effect of morphology. *Journal of The Electrochemical Society*, 167:013546, 2020.
- [20] I. V. Thorat, D. E. Stephenson, N. A. Zacharias, K. Zaghbi, J. N. Harb, and D. R. Wheeler. Quantifying tortuosity in porous Li-ion battery materials. *Journal of Power Sources*, 188:592–600, 2009.
- [21] N. A. Dodgson. Quadratic interpolation for image resampling. *IEEE Transactions on Image Processing*, 6:1322–1326, 1997.
- [22] L. Holzer, P. Marmet, M. Fingerle-Straß, A. Wiegmann, M. Neumann, and V. Schmidt. *Tortuosity and Microstructure Effects in Porous Media: Classical Theories, Empirical Data and Modern Methods*. Springer, 2023.
- [23] M. Osenberg, A. Hilger, M. Neumann, A. Wagner, N. Bohn, J. R. Binder, V. Schmidt, J. Banhart, and I. Manke. Classification of FIB/SEM-tomography images for highly porous multiphase materials using random forest classifiers. *Journal of Power Sources*, 570:233030, 2023.
- [24] A. Wagner, N. Bohn, H. Gesswein, M. Neumann, M. Osenberg, A. Hilger, I. Manke, V. Schmidt, and J. Binder. Hierarchical structuring of NMC111-cathode materials in lithium-ion batteries: An in-depth study on the influence of primary and secondary particle sizes on electrochemical performance. *ACS Applied Energy Materials*, 3:12565–12574, 2020.
- [25] M. Neumann, T. Philipp, M. Häringer, G. Neusser, J. R. Binder, and C. Kranz. Stochastic 3D modeling of nanostructured NVP/C active material particles for sodium-ion batteries. *Batteries & Supercaps*, 7:e202300409, 2024.
- [26] J. N. Reddy. *An Introduction to the Finite Element Method*. McGraw-Hill Education, 3rd edition, 2005.
- [27] S. C. Brenner and L. R. Scott. *The Mathematical Theory of Finite Element Methods*. Texts in Applied Mathematics. Springer, 3rd edition, 2008.
- [28] P. Sarangi, R. Hattori, T. Komiyama, and P. Pal. Super-resolution with binary priors: Theory and algorithms. *IEEE Transactions on Signal Processing*, 71:3213–3228, 2023.
- [29] T. Peleg and M. Elad. A statistical prediction model based on sparse representations for single image super-resolution. *IEEE Transactions on Image Processing*, 23:2569–2582, 2014.
- [30] H. He and W.-C. Siu. Single image super-resolution using Gaussian process regression. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 449–456, Colorado Springs, CO, USA, 2011.
- [31] W. Dong, L. Zhang, G. Shi, and X. Li. Nonlocally centralized sparse representation for image restoration. *IEEE Transactions on Image Processing*, 22:1620–1630, 2013.

- [32] M. Elad and A. Feuer. Restoration of a single superresolution image from several blurred, noisy, and undersampled measured images. *IEEE Transactions on Image Processing*, 6:1646–1658, 1997.
- [33] S. Farsiu, D. Robinson, M. Elad, and P. Milanfar. Advances and challenges in super-resolution. *International Journal of Imaging Systems and Technology*, 14:47–57, 2004.
- [34] R. B. Bird, W. E. Stewai, and E. N. Lightfoot. *Transport Phenomena*. J. Wiley & Sons, revised 2nd edition, 2007.
- [35] D. J. Griffiths. *Introduction to Electrodynamics*. Cambridge University Press, 2023.
- [36] D. Hlushkou, H. Liasneuski, U. Tallarek, and S. Torquato. Effective diffusion coefficients in random packings of polydisperse hard spheres from two-point and three-point correlation functions. *Journal of Applied Physics*, 118:124901, 2015.
- [37] J. Joos, T. Carraro, M. Ender, B. Ruger, A. Weber, and E. Ivers-Tiff e. Detailed microstructure analysis and 3D simulations of porous electrodes. *ECS Transactions*, 35:2357, 2011.
- [38] F. Willot, B. Abdallah, and Y.-P. Pellegrini. Fourier-based schemes with modified green operator for computing the electrical response of heterogeneous media with accurate local fields. *International Journal for Numerical Methods in Engineering*, 98:518–533, 2014.
- [39] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 105–114, Honolulu, HI, USA, 2017. IEEE Computer Society.
- [40] O. Furat, D. Finegan, Z. Yang, T. Kirstein, K. Smith, and V. Schmidt. Super-resolving microscopy images of Li-ion electrodes for fine-feature quantification using generative adversarial networks. *npj Computational Materials*, 8:68, 04 2022.
- [41] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.
- [42] W. Shi, J. Caballero, F. Huszar, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1874–1883, Las Vegas, NV, USA, 2016.
- [43] V. Nair and G. E. Hinton. Rectified linear units improve restricted Boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning*, page 807–814, Madison, WI, USA, 2010.
- [44] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, San Diego, CA, USA, 2015.
- [45] J. A. Nelder and R. Mead. A simplex method for function minimization. *The Computer Journal*, 7:308–313, 1965.
- [46] P. Soille. *Morphological Image Analysis: Principles and Applications*. Springer, 2nd edition, 2004.
- [47] M. Neumann, C. Hirsch, J. Staněk, V. Beneš, and V. Schmidt. Estimation of geodesic tortuosity and constrictivity in stationary random closed sets. *Scandinavian Journal of Statistics*, 46:848–884, 2019.

## APPENDIX

Figure A1 shows exemplary 2D slices of super-resolved tomographic image data together with the corresponding ground truth image. Figure A2 shows the relative errors of mean geodesic and effective tortuosity, separately for the solid and pore phase, as additional information to Figure 6 of the main text.

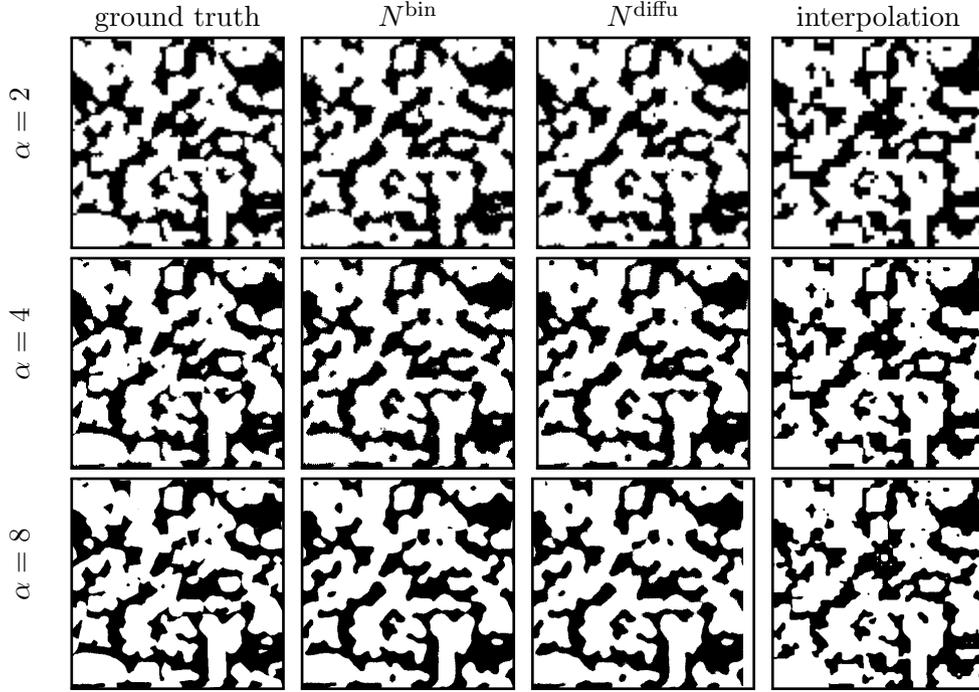


FIGURE A1. Exemplary 2D slices of super-resolved tomographic image data, obtained by means of a given method (columns) and at a given scaling factor  $\alpha$  (rows) together with the corresponding ground truth. This sample consists of medium-grained primary particles that have been sintered at  $900^\circ\text{C}$ .

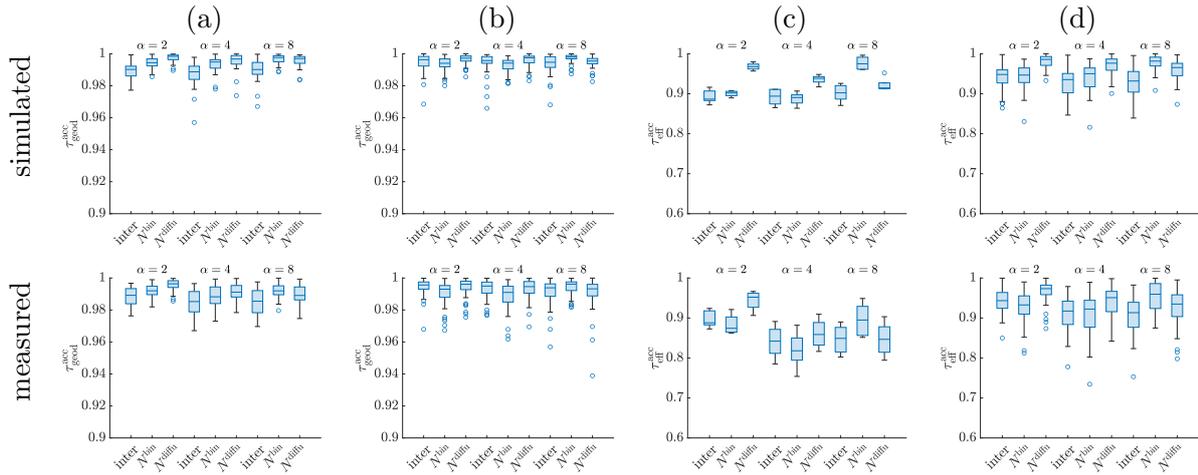


FIGURE A2. Relative errors of mean geodesic tortuosity and effective tortuosity, separately plotted for pore and solid phase. Top row: Simulated image data: Bottom row: Tomographic image data. Shown are the relative errors of mean geodesic tortuosity for solid phase (a) and pore phase (b), as well as the relative errors of effective tortuosities for solid phase (c) and pore phase (d).