Conditional generative AI for high-fidelity synthesis of hydrating cementitious microstructures

Liang, Minfei; Feng, Kun; Xie, Jinbao; Wei, Yuyang; Contera, Sonia; Schlangen, Erik; Šavija, Branko

DOI
10.1016/j.matdes.2025.114251

Publication date
2025

Document Version
Final published version

Published in
Materials and Design

Materials & Design 256 (2025) 114251

Contents lists available at ScienceDirect

Materials & Design

journal homepage: www.elsevier.com/locate/matdes

Conditional generative AI for high-fidelity synthesis of hydrating cementitious microstructures

Minfei Liang [a,b], Kun Feng [c,*], Jinbao Xie [a], Yuyang Wei [d], Sonia Contera [b], Erik Schlangen [a], Branko Šavija [a,*]

[a] Microlab, Faculty of Civil Engineering and Geosciences, Delft University of Technology, Delft 2628CN, the Netherlands
[b] Clarendon Laboratory, Department of Physics, University of Oxford, Oxford OX1 3PU, UK
[c] Key Laboratory of Transportation Tunnel Engineering, Ministry of Education, Southwest Jiaotong University, Chengdu 610031 Sichuan, China
[d] Department of Engineering Science, University of Oxford, Oxford OX1 3PJ, UK

ARTICLE INFO

Keywords:
Cement paste
Generative AI
Microstructure
Hydration
Transformer

ABSTRACT

Portland cement paste has a highly heterogenous evolving microstructure that complicates the development of stronger and greener cementitious materials. Microstructure is the fundamental input of multiscale studies on material behaviors. Herein, we propose a conditional generative AI framework for synthesizing high-fidelity 3D microstructures of hydrating cement paste (1–28 days) with varying water-to-cement ratios and Blaine fineness values. A latent diffusion transformer, operating within a compact two-stage latent space derived via a vector quantized variational autoencoder, efficiently captures and reproduces experimentally measured microstructural patterns. Statistical analyses confirm strong consistency in grey value distributions, micromechanical properties, hydration phase evolution, and particle size distributions, with only minor boundary-related discrepancies. Validation using a pretrained classifier further corroborates the fidelity of generated microstructures. This approach provides a robust tool for realistic cement paste microstructure generation, supporting multiscale modeling and advancing the design of sustainable cementitious materials.

1. Introduction

Portland cement paste is the most widely used binder in construction materials, with its production contributing over 2.7 billion tons of CO_2 emissions annually [1]. Despite extensive application, a thorough understanding of cement paste is still lacking because of its highly heterogenous and time-evolving microstructure. Such microstructural complexity presents a challenge to advancing fundamental research on material properties and development/design of durable and green binders [2].

Microstructural analysis offers critical insights into the intrinsic properties of cementitious materials [3]. Using a realistic microstructure as input, multiscale computational models—such as Finite Element Method (FEM) [4,5], micromechanical homogenization schemes [6–8], Lattice Fracture Models [9,10]—can be built to establish microstructure-property relationships. Typically, the microstructure of cement paste is characterized by experiments using scanning electron microscopy (SEM) and X-ray computed tomography (XCT) [11–13], which demand advanced instrumentation and delicate sample preparation [14]. Alternatively, researchers have turned to simulating hydration reaction to obtain microstructures, leading to the development of hydration models such as Hymostruc [15–17], μic [18], CemHyd3D [19], and HYD-NSP [20]. These models, validated through experiments, demonstrate reasonable hydration kinetics and generate complex microstructural formations. However, the intricate thermodynamics of cement hydration—driven by multifield physics, diverse environmental factors, reactants and products—necessitates numerous assumptions, such as empirical coefficients and simplified particle shapes, to manage model complexity.

The challenges faced by the traditional approaches mentioned above underscore the difficulty of efficiently capturing realistic cement paste microstructures. From a statistical perspective, the microstructure is essentially a joint distribution of pixel values over the geometry space. Low-order probability functions [21] have been applied to approximate the joint distribution of concrete microstructure and calculate the properties such as permeability, stiffness, and thermal conductivity

* Corresponding authors.
E-mail addresses: minfei.liang@physics.ox.ac.uk (M. Liang), windfeng813@163.com (K. Feng), J.Xie-1@tudelft.nl (J. Xie), yuyang.wei@eng.ox.ac.uk (Y. Wei), sonia.antoranzcontera@physics.ox.ac.uk (S. Contera), Erik.Schlangen@tudelft.nl (E. Schlangen), B.Savija@tudelft.nl (B. Šavija).

https://doi.org/10.1016/j.matdes.2025.114251
Received 26 February 2025; Received in revised form 15 April 2025; Accepted 15 June 2025
Available online 16 June 2025

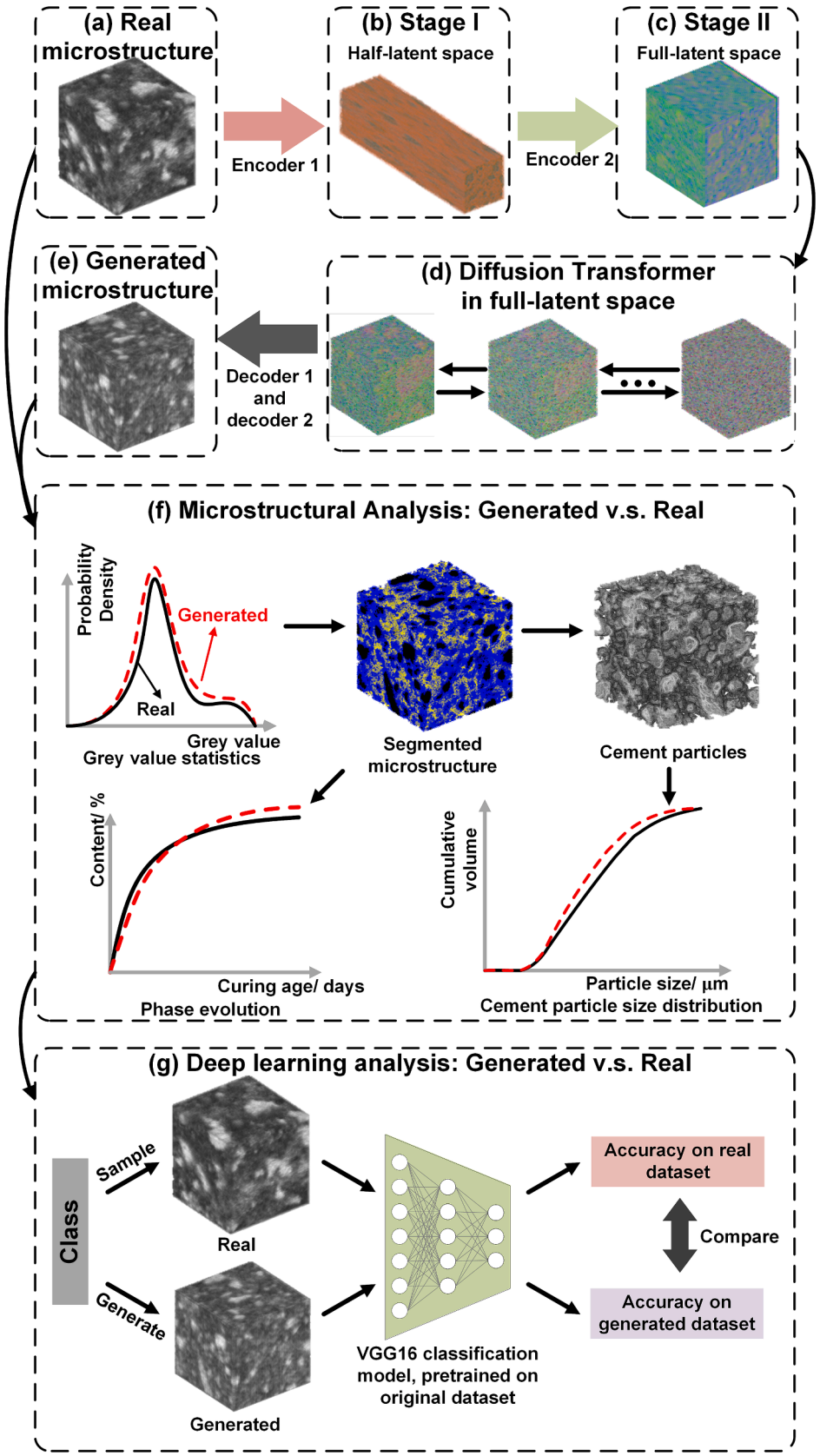


Fig. 1. Overall workflow of TL-DiT: Using VQVAE, the 192 × 192 × 192 × 1 microstructure **(a)** is first encoded to 48 × 48 × 192 × 3 half-latent in stage I **(b)**, and then 48 × 48 × 48 × 3 full-latent in stage II **(c)**. A diffusion transformer **(d)** models the full-latent representation, which can then be sequentially decoded by VQVAE in state I and II to obtain the generated microstructure **(e)**. A series of microstructural analysis are conducted to compare the statistical and morphological difference between generated and real microstructures **(f)**. The pretrained VGG-16 model is used to compare the generated and real microstructures **(g)**.

[22,23]. However, high computational costs and oversimplified phase assumptions were noted, causing discrepancy in microstructural morphology and macroscopic properties [24].

In recent years, the generative deep learning models were successfully applied to capture the complex distribution function of various microstructures [25]. Among them, variational autoencoder (VAE) [26] and generative adversarial network (GAN) [27] were shown to be effective in generating microstructures/metastructures of various materials, such as applications of VAE for two-phase steels and lattice metamaterials [28,29] and GAN for carbon steel and heterogenous energetic materials [30–32]. Recently, Hong et al. [33] applied GAN for synthesizing microstructure of cement paste which however is subjected to four-phase assumptions. Although promising results were obtained, limitations have been noted, such as the blurry images induced by Gaussian prior/posterior assumptions in VAEs [34] and unstable training and mode collapse induced by the adversarial training in GANs [35,36].

Denoising Diffusion Probabilistic Models (DDPM) [37] have recently proven effective for microstructure generation [38], using a predefined forward diffusion and neural network-based reverse denoising process. This approach overcomes the limitations of VAEs' single Gaussian prior [39] and generates images with greater quality and variability than GANs [40]. Successful applications of DDPMs are noted in generating microstructures of magnesium alloy [41], fiber composite [42], multifunctional composites [43], and cementitious materials [44]. These studies showed that DDPM can generate microstructures that demonstrate similar statistical and physical patterns. Nevertheless, these studies were generally performed on 2D datasets, presumably due to the limitations of computational resources and scarcity of comprehensive 3D datasets. Alternatively, Lee et al. [45,46] developed a multi-plane DDPM model, expanding dimensionality to enable 2D-to-3D microstructure reconstruction. Such multi-plane training strategies are similar to SliceGAN [47], which also generates 3D microstructures from 2D images. Despite producing realistic microstructures, these methods capture primarily 2D slice-level information while overlooking depth-wise details.

To meet the demand for high-fidelity 3D microstructure generation of cement paste and overcome challenges in current DDPM applications, this paper proposes a Two-stage Latent Diffusion Transformer (TL-DiT) to synthesize 3D microstructure of cement paste. The model introduces a two-stage latent representation approach to efficiently compress 3D datasets while preserving key microstructural information. A vision transformer capable of capturing slice-depth information forms the backbone of the DDPM in latent space. This model can generate microstructures of cement paste with varying water-to-cement ratios (w/c), Blaine values, and curing ages, resulting in outputs that not only visually resemble real microstructures but also align closely in statistical, microstructural, and morphological analyses. These findings indicate that the proposed model can generate high-fidelity microstructural inputs, suitable for further fundamental studies to establish microstructure-property relationships.

2. Overall workflow

The overall workflow of the TL-DiT is shown in Fig. 1. Inspired by latent diffusion framework [48], this study first encodes the microstructure into the latent space, models the distribution of latent representation, and decodes the latent into pixel space to obtain generated microstructure. A two-stage strategy is adopted to approach the latent space. The $192 \times 192 \times 192 \times 1$ grayscale microstructure (Fig. 1(a)) is first projected into $48 \times 48 \times 192 \times 3$ half latent in the stage I (Fig. 1(b)) and then $48 \times 48 \times 48 \times 3$ full latent in the stage II (Fig. 1(c)). Two vector quantized variational autoencoders (VQVAEs) [49] are used to bridge the pixel space, half latent space, and full latent space. The backbone of the VQVAE is a convolutional UNet [50] architecture, aiming to autoregressively learn the discrete latent representation and model their distributions. The perceptual and adversarial loss are incorporated to guide the training of VQVAE for reservation of essential and detailed microstructural information [51]. The perceptual loss is derived by a VGG-16 [52] pretrained on the real microstructure data. The adversarial loss involves a convolutional neural network as the discriminator and the VQVAE as the generator. The overall architecture of the TL-DiT can be found in Appendix A.

In the full latent space, a diffusion transformer [53,54] is used to model the distribution of the latent representation (Fig. 1(d)), which can then be sequentially decoded by the two VQVAEs to obtain microstructure (Fig. 1(e)). The transformer backbone comprises slice-wise block and depth-wise block to capture the 3D microstructural information. Each transformer block consists of a multi-head attention layer and a multi-layer perceptron (MLP) layer. Within each transformer block, Adaptive layer normalization [53–55] is adopted to process the diffusion time steps and class conditions of cement paste (i.e., w/c ratios, Blaine values, and curing ages), which enables conditional generation.

Using the established TL-DiT, a database of generated microstructures is constructed. Using generated and real microstructures as input, a series of microstructural analyses is conducted to compare their statistical and morphological differences (Fig. 1(f)). In addition, the established VGG-16 model (pretrained in stage I) is leveraged to conduct classification tasks on original and generated dataset. The confusion matrix of both classification tasks is analyzed and compared to show the high-level perceptual difference between the real and generated microstructures (Fig. 1(g)).

3. Dataset overview and data processing

The real microstructures was sourced from the open-access dataset of digitized 3D hardened cement paste microstructures obtained from XCT [13]. In this dataset, four types of ordinary Portland cement paste were used, varying in w/c ratios (w/c = 0.35 and 0.50), and Blaine values (Blaine = 273 m^2/kg and 391 m^2/kg). The XCT images were taken at 1, 2, 3, 4, 7, 14, and 28 days after hydration under 20 °C and fully saturated conditions. The specimens used in the XCT tests are cylindrical with both diameter and height of around 1 mm. The imaging resolution (i.e., voxel size) is around 1.1 μm.

For each of the 28 parameter combinations, 256 cubic samples are first cropped with the size $200 \times 200 \times 200$ voxels from each of the cylinder, establishing a dataset containing 7168 microstructural cubes. the microstructural cubes are resized into $192 \times 192 \times 192$ voxels based on the SciPy library [56] with linear interpolation. Therefore, the size of studied microstructure is around $210 \times 210 \times 210$ μm^3, with a voxel size of around 1.09 μ m (i.e., 210/192).

For training the VQVAE in stage I, 12 192×192 slices are extracted from each cube by selecting 4 evenly spaced slices along each of the three axes, resulting in a dataset of 81,016 slices for training. Then, for the VQVAE in stage II, the VQVAE in stage I is applied to all the microstructural cubes slice-by-slice and obtained the prismatic half-latent with the size $48 \times 48 \times 192$ for each cube. Along the two axes with the size of 48, 12 192×48 slices are extracted by selecting 6 evenly spaced slices, resulting in also a dataset of 81,016 slices for training. Finally, the two VQVAEs in stage I and II are used to compress the $192 \times 192 \times 192$ cubes into $48 \times 48 \times 48$ full latent, which can then be used to train the diffusion transformer model. During each stage, the data is normalized to (0, 1) with min–max normalization and then scaled to (−1, 1) for training.

4. Two-stage latent diffusion transformer

4.1. Vector-quantitated variational autoencoder

As shown in Fig. 1 (a), two VQVAEs are used to derive the two-stage latent representation. Both VQVAEs use convolutional UNet architecture as the backbone to autoregressively conduct data compression and

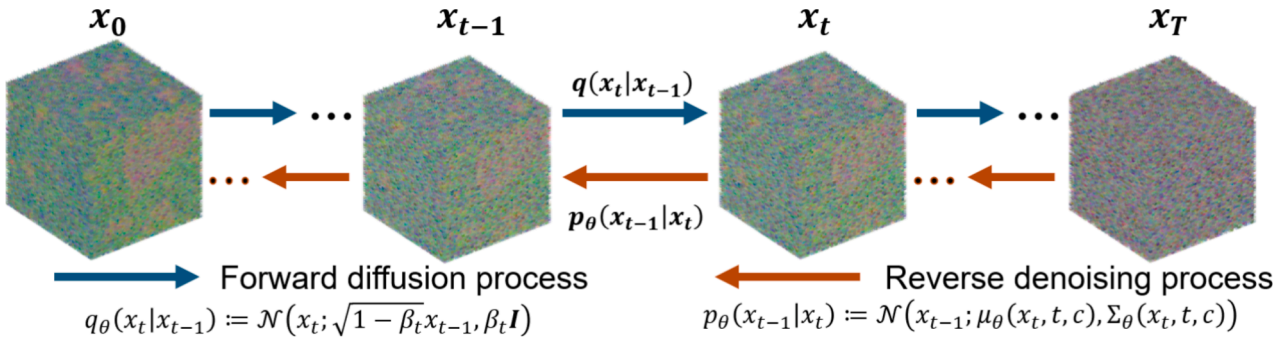


Fig. 2. Forward diffusion and reverse denoising process of DDPM.

reconstruction. Details of the architecture and the training process can be found in Appendix B. The VQVAE is trained in an unsupervised manner and uses the loss function as below:

$$L_{VAE} = L_{recon} + L_{code} + \lambda_{com} L_{com} + \lambda_{pe} L_{pe} + \lambda_{ad} L_{ad} \tag{1}$$

where the total loss L_{VAE} is the sum of reconstruction loss L_{recon}, codebook loss L_{code}, commitment loss L_{com}, and weighted loss of perceptual loss L_{pe} and adversarial loss L_{ad}. λ_{com}, λ_{pe} and λ_{ad} represent the wights of commitment, perceptual and adversarial loss term, which are 0.2, 1 and 0.8 in this study. The reconstruction loss, codebook loss and commitment loss are defined as the real VQVAE paper [49]. The perceptual and adversarial loss are introduced as below:

- The perceptual loss is quantified by the metric Learned Perceptual Image Patch Similarity (LPIPS) [57], which is calculated from the feature maps of a pretrained VGG-16 classification model [58]. To ensure that the VGG-16 model can extract essential information from the microstructure, the VGG-16 model is pretrained with a classification task, where the real microstructural slice was used as input and the class information —w/c, Blaine values, and curing ages— was the output. A combination of cross entropy functions is used as the loss function to ensure that the prediction accuracy of the VGG-16 classification reaches almost 100 % in a dataset of 81,016 microstructural slices. The details of the perceptual loss part can be found in Appendix B.2.
- The adversarial loss is calculated during the adversarial training between the VQVAE generator and a discriminator built with sequential convolutional networks. The discriminator assesses the realism of small patches within the microstructural slice input, encouraging the VQVAE generator to focus on fine-grained features and preserve high-frequency details [59]. The details of the adversarial loss part can be found in Appendix B.3.

4.2. Latent diffusion transformer

4.2.1. Vision transformer architecture

Using the trained VQVAEs, all the 3D microstructural cubes are projected into full latent spaces. Then a latent diffusion transformer is employed to model the distribution of these latent representations, enabling us to ultimately decode them back into microstructures using the VQVAEs.

A vision transformer is used as the backbone of the latent diffusion transformer. The architecture of the vision transformer is explained in Appendix A. Initially, the full 48 × 48 × 48 latent representation is divided into patches of size 4 × 4 × 4, with each patch subsequently embedded into a token with a hidden size of 768. Note that a fine patch size is used here to capture the fine-grained feature. To effectively capture 3D microstructural information, we use alternating slice-depth transformer blocks. This model comprises 12 transformer blocks, each incorporating attention layers with 4 attention heads.

To incorporate conditioning, both the class information for the cement microstructure and diffusion time steps are first embedded as tokens of size 768. These conditioning tokens are then passed to each transformer block, where they are integrated into the microstructure tokens through adaptive layer normalization, supporting dynamic conditioning throughout the model. The resulting vision transformer has a total of 129,734,832 parameters.

4.2.2. Denoising diffusion probabilistic model

DDPM [37] operates through two sequential stages (Fig. 2): a forward diffusion phase, where noise is progressively added to an image, and a reverse denoising phase, where the model gradually removes this noise to reconstruct the original image. Both stages are Markov processes, meaning that each step only depends on the previous one. In the forward diffusion stage, noise is incrementally introduced to the original image x_0 over a series of steps t = 0 to T by applying a predefined transition function $q(x_t|x_{t-1})$. By the final step, the image is indistinguishable from standard Gaussian noise. During the reverse denoising phase, vision transformers are used to approximate the reverse transition function $p_\theta(x_{t-1}|x_t)$ and iteratively remove noise at each step. Both $q(x_t|x_{t-1})$ and $p_\theta(x_{t-1}|x_t)$ are assumed to follow Gaussian distributions.

In essence, DDPM can be conceptualized as a hierarchical form of a VAE. However, while a traditional VAE approximates the data distribution through a single Gaussian distribution, DDPM iteratively refines its predictions through multiple steps of Gaussian noise removal. This hierarchical structure allows DDPM to better accommodate the Gaussian assumption in its transitions without significantly compromising its modeling capacity.

The noise-adding step in the forward diffusion process is defined as:

$$q(x_t|x_{t-1}) = N\left(x_t; \sqrt{1-\beta_t}\, x_{t-1}, \beta_t \boldsymbol{I}\right) \tag{2-1}$$

$$q(x_{1:T}|x_0) = \prod_{t=1}^{T} q(x_t|x_{t-1}) \tag{2-2}$$

where β_t is a predefined noise schedule that gradually increases from 0.0001 to 0.02 over t = 0 to 1000 steps. Using the additive properties of Gaussian distributions, the noisy image at any step can be written directly in terms of x_0 and β_t, as below:

$$q(x_t|x_0) = N(x_t; \sqrt{\overline{\alpha}_t}\, x_0, (1-\overline{\alpha}_t)\boldsymbol{I}) \tag{3}$$

where $\alpha_t = 1 - \beta_t$, $\overline{\alpha}_t = \prod_{i=0}^{t} \alpha_i$. By applying Bayes' rule to the forward diffusion process and utilizing Eq. (2 ~ 3), the posterior distribution is obtained:

$$q(x_{t-1}|x_t) = N\left(x_{t-1}; \widetilde{\mu}_t(x_t, x_0), \widetilde{\beta}_t \boldsymbol{I}\right) \tag{4-1}$$

$$\widetilde{\mu}_t(x_t, x_0) = \frac{\sqrt{\overline{\alpha}_{t-1}}\beta_t}{1-\overline{\alpha}_t} x_0 + \frac{\sqrt{\alpha_t}(1-\overline{\alpha}_{t-1})}{1-\overline{\alpha}_t} x_t \tag{4-2}$$

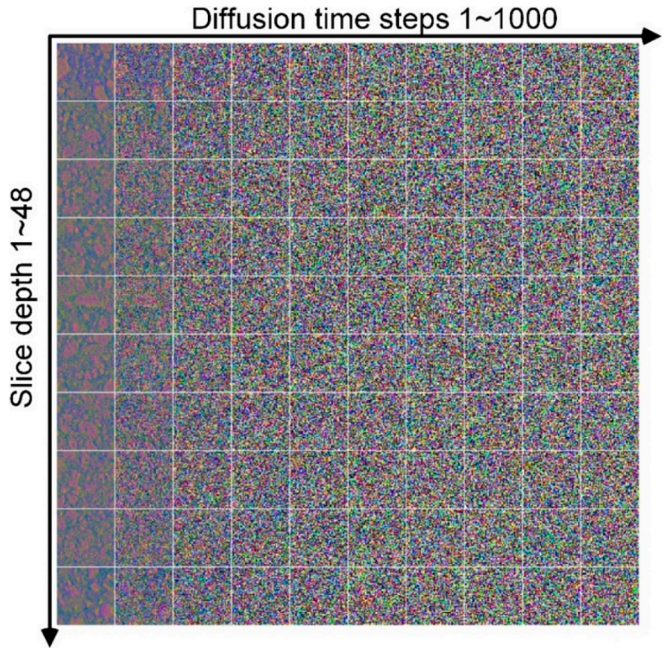


Fig. 3. Slices of a noised latent cube at different diffusion time steps.

$$\widetilde{\beta}_t = \frac{1-\overline{\alpha}_{t-1}}{1-\overline{\alpha}_t}\beta_t \tag{4-3}$$

The transition function for the reverse denoising phase is also a Gaussian:

$$p_\theta(x_{t-1}|x_t) = N\left(x_{t-1}; \mu_\theta(x_t,t), {\sigma_t}^2 \boldsymbol{I}\right) \tag{5}$$

where ${\sigma_t}^2$ can be assumed the same as β_t or $\widetilde{\beta}_t$, which both lead to similar results [37]. In this case, we use${\sigma_t}^2 = \widetilde{\beta}_t$. The loss function for DDPM training is derived from maximizing the evidence lower bound (ELBO) on the negative log-likelihood. Similar to a VAE, this training objective encourages the model to maximize the likelihood of the data by minimizing:

$$\begin{aligned}\mathbb{E}[-\log p_\theta(x_0)] &\le \mathbb{E}_q\left[-\log\frac{p_\theta(x_{0:T})}{q(x_{1:T}|x_0)}\right] \\ &= \mathbb{E}_q\left[-\log p(x_T) - \sum_{t\ge 1}\log\frac{p_\theta(x_{t-1}|x_t)}{q(x_t|x_{t-1})}\right]\end{aligned} \tag{6}$$

Rearranging terms leads to the loss function:

$$L = \mathbb{E}_q[D_{KL}(q(x_T|x_0)\|p(x_T)) - \log p_\theta(x_0|x_1) + D_{KL}(q(x_{t-1}|x_0,x_t)\|p_\theta(x_{t-1}|x_t))] \tag{7}$$

where D_{KL} is the Kullback–Leibler divergence [60], which is a common measure of distance between two probability distributions and can be calculated by $D_{KL}(p(x)\|q(x)) = \int p(x)\log\frac{p(x)}{q(x)}dx$. The first term in Eq (7) is a constant term and can be ignored, since both $q(x_T|x_0)$ and $p(x_T)$ are standard Gaussian distribution. The second term is the reconstruction term. The last term is the distance between the posterior of the forward diffusion (Eq (4)) and the transition in the reverse process (Eq (5)). Because we set ${\sigma_t}^2 = \widetilde{\beta}_t$, minimizing the last term is equivalent to fitting $\mu_\theta(x_t,t)$ with $\widetilde{\mu}_t(x_t,x_0)$. By reparametrizing the second and third term following Eq (4), a simplified loss function for the DDPM can be obtained, as below:

$$L_{sym} = \mathbb{E}_{t,x_0,\epsilon}\left[\|\epsilon - \epsilon_\theta\left(\sqrt{\overline{\alpha}_t}x_0 + \sqrt{1-\overline{\alpha}_t}\epsilon, t\right)\|^2\right] \tag{8}$$

where $\epsilon_\theta\left(\sqrt{\overline{\alpha}_t}x_0 + \sqrt{1-\overline{\alpha}_t}\epsilon\right)$ is the vision transformer which takes the original images x_0 and step t as the input and predicts the stepwise noise that needs to be removed; ϵ is the noise sampled from a standard Gaussian distribution in the forward diffusion process.

4.2.3. Training

The trained VQVAEs in stage I and II are applied to project the real microstructures into the full latent space, and then the diffusion transformer is trained to model the distribution of the full latent, which can then be decoded by the VQVAEs to generated microstructures. The diffusion transformer is trained in mini-batches with a batch size of 32 for a total of 200,000 training steps. The training is conducted on two NVIDIA RTX 6000 GPUs (Ada architecture), each with 48 GB of memory. The entire training process takes approximately 25 h. The maximum step T of the forward diffusion process is 1000. In the training process, the time step t is sampled from a uniform distribution of 1 ~ 1000, and the noise ε is sampled from a standard Gaussian distribution. Using the original image x_0, time step t, and the noise ε as input, the diffusion process (Eq (2)) can be conducted to get the noised image x_t. Slices of a noised latent cube at equally spaced time steps from 1 ~ 1000 and depth from 1 ~ 48 are shown in Fig. 3.

The transformer can then take the noised image x_t, time step t, and class conditions including w/c ratios, Blaine values, and curing ages, as the input to predict the noise, ε_θ for the reverse denoising process. Using the gradient descent optimizer Adam [61], the parameters of the vision transformer (i.e., θ) are optimized according to the loss function in Eq

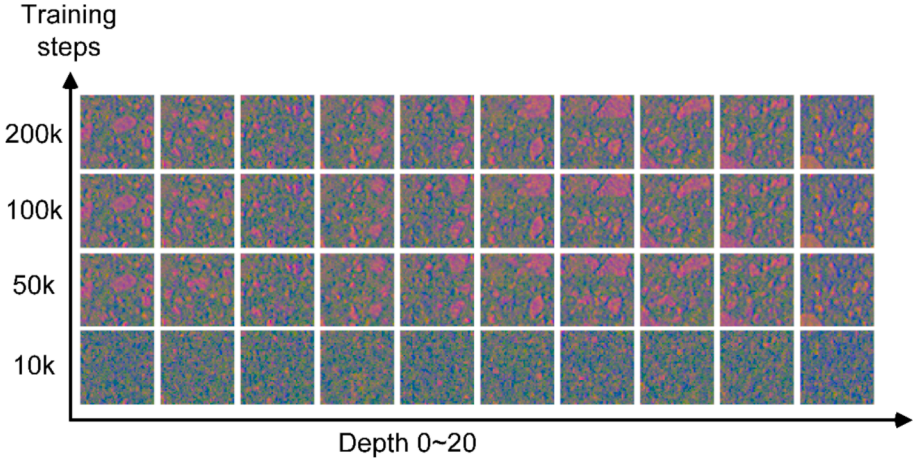


Fig. 4. Slices of a generated latent cube at different depths.

Table 1
Parameter settings of the hydration kinetics model.

Blaine value	A_1	A_2	η
273	6×10^7	5×10^{-2}	8.5
391	8×10^7	8×10^{-2}	8

(8). In the gradient descent optimization, a cosine warm-up schedule for the learning rate is used, which comprises a linear warm-up in the first 10,000 steps and a cosine-shaped decay afterwards as done in many large language models to improve convergence [62]. Fig. 4 presents the slices of a latent cube generated from the same standard Gaussian noise at different training steps. The training results show that the general pattern of the latent cube stabilizes before first 50,000 steps, and afterwards the model only finetunes the output, resulting in similar results.

5. Microstructural analysis

5.1. Hydration model for microstructural segmentation

We use an established hydration model to estimate the content of each phase, which then serves as a reference for image-based phase segmentation, as also adopted by the dataset authors [63].

5.1.1. Hydration model

The hydration model includes two parts: an Arrhenius equation for solving the hydration kinetics and a series of empirical models for establishing the relationship between hydration degrees and phase contents. Assuming that the hydration kinetics can be described by postulating the existence of a Gibb's free energy dependent on temperature and hydration extent, the hydration reaction can be described by the following Arrhenius equation [64–66]:

$$\dot{\alpha} = A(\alpha)e^{-\frac{E_{ac}}{RT}} \tag{9-1}$$

$$A(\alpha) = A_1\left(\frac{A_2}{\alpha_{ult}} + \alpha\right)(\alpha_{ult} - \alpha)e^{-\eta\frac{\alpha}{\alpha_{ult}}} \tag{9-2}$$

$$\alpha_{ult} = \frac{1.031w/c}{0.194 + w/c} \tag{9-3}$$

where α is the hydration degree; E_{ac} is the apparent activation energy; R is the universal gas constant (8.314 J/mol·K); T is the temperature, which is 293 K in this study; η, A1, and A2 are fitting parameters corresponding to a certain type of cement, which can be calibrated according to adiabatic test results [65,67]; αult is the ultimate hydration degree dependent on the water cement ratio w/c. This paper assumes that the cement with Blaine fineness values of 273 and 391 correspond to normal cement (32.5R or 42.5 N) and fast hardening cement (42.5R, 52.5 N, 52.5R), respectively. Then, according to [65,67,68], the parameters given in Table 1 are used herein for the hydration kinetic calculations. The apparent activation energy of cement is assumed as a constant, which is 41570 J/mol [67].

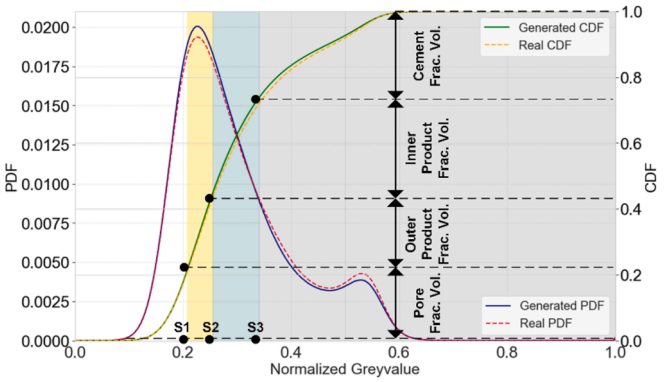


Fig. 6. Calculation of the threshold points for microstructural segmentation.

By solving Eq. (9), the evolution of hydration degrees is obtained for cement paste of each parameter setting. Then, the powers' model [69–71] is used to further calculate the evolution of porosity and unhydrated cement:

$$\varphi(\alpha) = \frac{\rho_{cem}(w/c) - f_{exp}\alpha}{1 + \rho_{cem}(w/c)} \tag{10-1}$$

$$\gamma(\alpha) = \frac{1 - \alpha}{1 + \rho_{cem}(w/c)} \tag{10-2}$$

where φ and γ are porosity and unhydrated cement content, respectively; ρ_{cem} is the specific gravity of cement (herein taken to be 3.2); f_{exp} is the volumetric expansion coefficient for the "solid" cement hydration products relative to the cement reacted (herein taken to be 1.4 [19,70]). Based on Eq(9 ~ 10), the phase evolution results can be obtained for the four types of cement paste, as shown in Fig. S12.

Based on results in Fig. 5, microstructure can be segmented into three phases: pores, hydration products, and unhydrated cement. Then, using Jenning's model [72], the hydration products can be further divided into inner product and outer product:

$$M_r = 3.017(w/c)\alpha - 1.347\alpha + 0.538 \tag{11}$$

where M_r is the ratio of the mass of outer product to inner product. Therefore, with the hydration model comprised by Eq (9 ~ 11), the microstructure can be segmented into a four-phase composite and then more in-depth comparison of generated and real microstructures can be

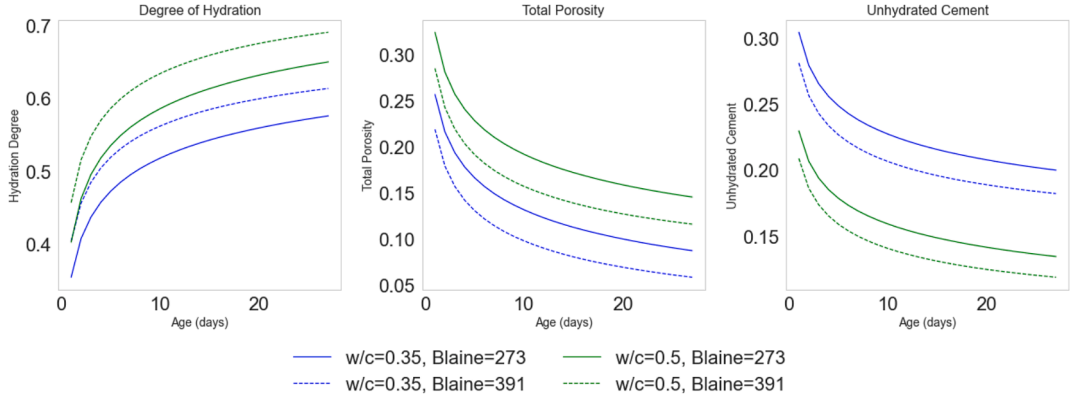


Fig. 5. Phase evolution of the four types of cement paste given by the hydration model.

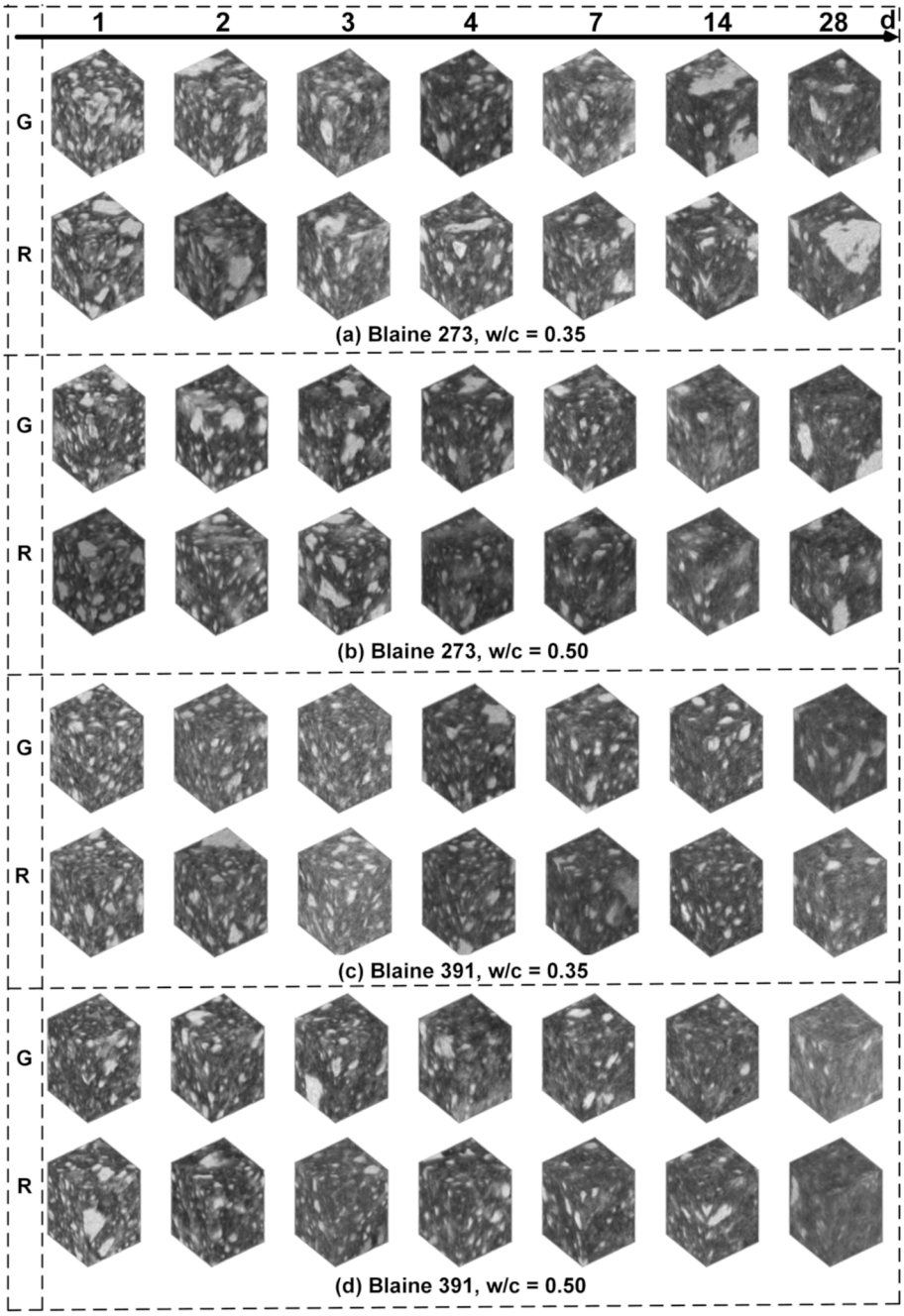


Fig. 7. Randomly selected generated and real microstructures of different w/c ratios, Blaine values, and curing ages (Note: G stands for "generated" samples and R stands for "Real" samples).

conducted.

5.1.2. Microstructure segmentation

Based on the estimated phase volume fractions derived from the hydration model, the microstructure is segmented into four primary phases: unhydrated cement, inner product, outer product, and pores. The segmentation is conducted by determining three thresholding points on the greyscale CDF of the microstructural image, as done by the authors of the dataset [63]. As shown in Fig. 6, and considering the fact that the brightness of unhydrated cement, inner product, outer product, and pores typically decreases in this order, the greyvalue threshold corresponding to unhydrated cement (i.e., S3) is first determined by subtracting the predicted volume fraction of cement from the upper limit of the CDF (i.e., 1.0). Subsequently, the thresholds for inner product (S2) and outer product (S1) are determined by sequentially subtracting their respective volume fractions from the remaining CDF interval.

The accuracy of such segmentation method essentially depends on the validity of the adopted hydration model, whose parameters in this study were calibrated to yield phase fractions comparable to those reported in the original work [63]. While alternative segmentation methods are available— such as those involving detailed analysis of local microstructural features and case-specific threshold selection, as shown in [73] —they often involve subjective tuning and are less suitable for comparative studies. This work does not aim to validate the hydration model, its parameters, or the segmentation methodology itself, but rather to establish a unified and consistent framework that enables meaningful comparison of the phase assemblages and their

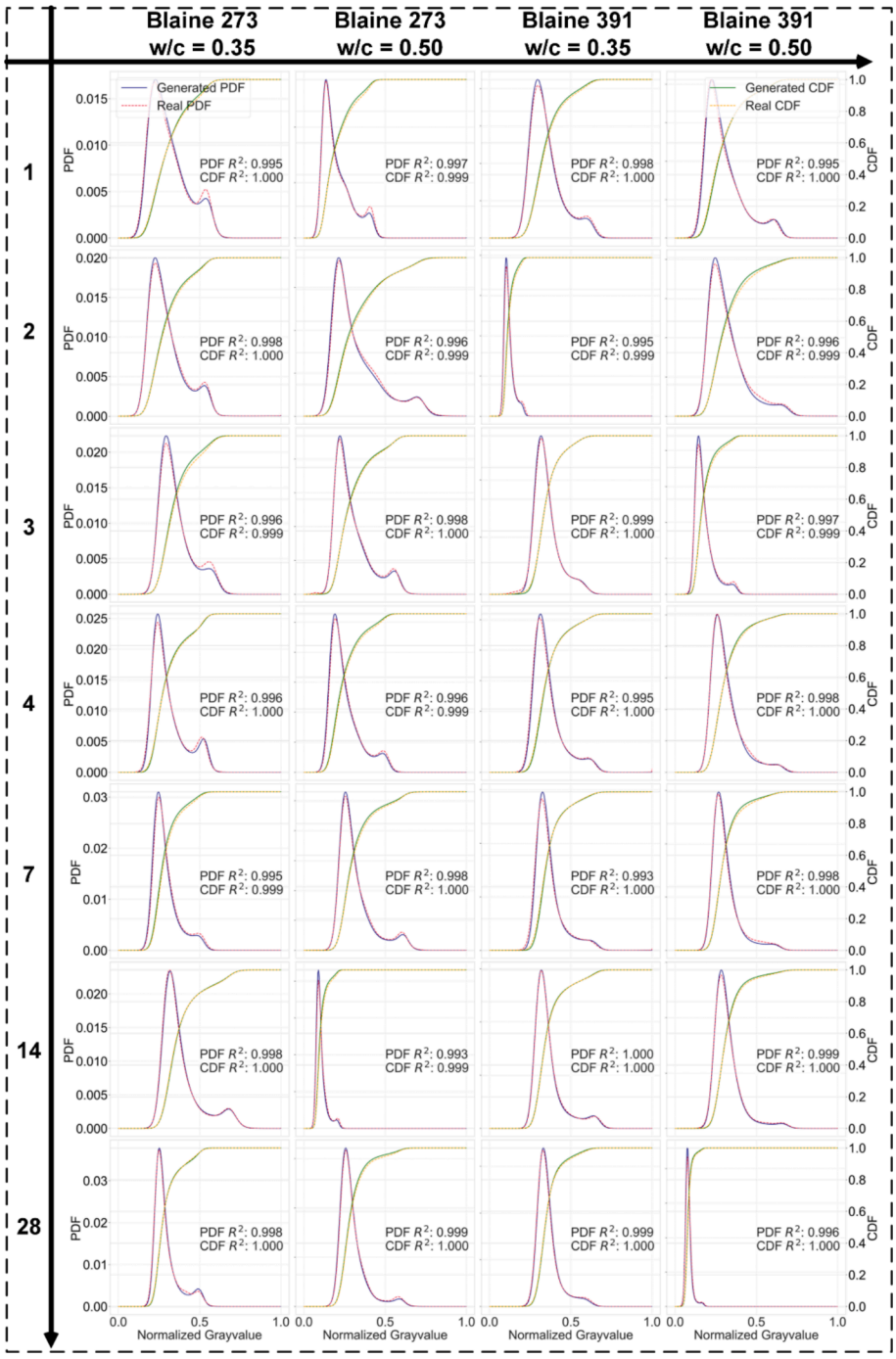


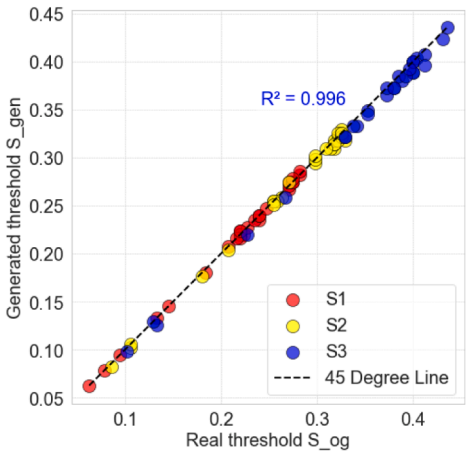
Fig. 8. Comparison between the probability distribution function (PDF) and cumulative distribution function (CDF) of the gray values of generated and real microstructures with different w/c ratios, Blaine values, and curing ages.

evolution between generated and real microstructures.

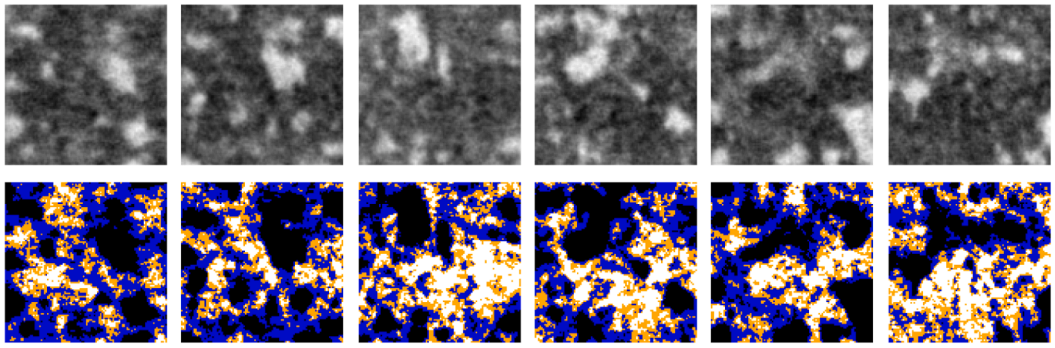
5.2. Size distribution of cement particles

This paper used the SciPy [56] and scikit-image [74] to calculate cement particle sizes. To analyze particle size distributions, 3D microstructural cubes were extracted from both real and generated datasets. For improved data quality, a preprocessing step using Total Variation Denoising (TVD) was applied, which minimizes noise while preserving edges in the microstructure. The denoised data cubes were then binarized based on predefined phase thresholds to isolate the target microstructural phase for further analysis.

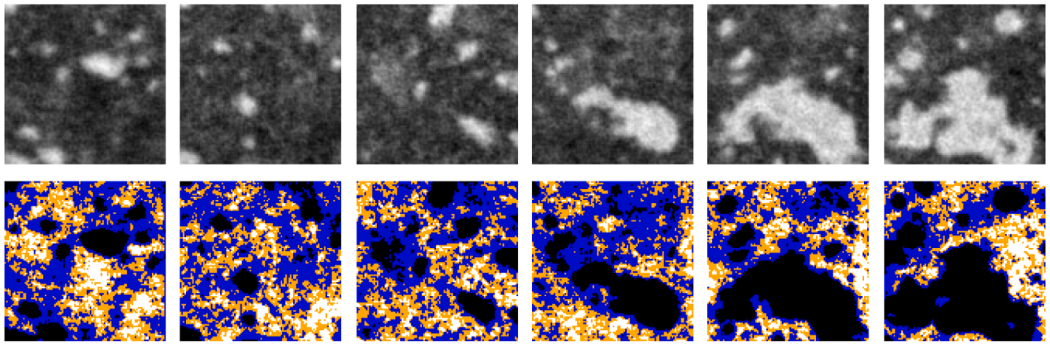
The particles within the microstructure were segmented using a watershed-based approach. First, a Euclidean distance transform was applied to the binary microstructure, generating a distance map where each voxel's value represents its distance to the nearest background voxel. Local maxima of this distance map were then identified as particle centers, constrained by a minimum distance threshold to mitigate the risk of over-segmentation. At early ages, unhydrated cement particles are typically larger and more densely clustered, increasing the likelihood that a single particle may be mistakenly split into multiple segments. To address this, a larger threshold was applied at early hydration stages, while smaller values were used at later ages as the particles become smaller and more dispersed. Specifically, the minimum distance



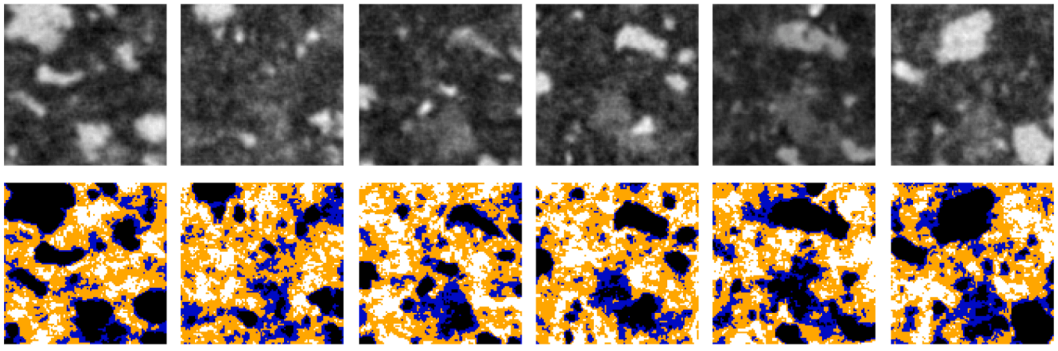
(a)



(b) w/c = 0.35, Blaine values = 273, curing age = 7 days



(c) w/c = 0.35, Blaine values = 273, curing age = 28 days



(d) w/c = 0.50, Blaine values = 273, curing age = 7 days

Fig. 9. Microstructural phase segmentation: based on the CDF graph, the threshold values S1, S2, S3 can be calculated based on volume fraction of each phase calculated by a hydration model. The threshold points (a) of generated and real microstructures over 28 types of parameter combinations are clustered near the 45-degree line, indicating strong similarity; Six slices (b ~ e) are cropped (depth-wise) every 19 μm from the generated microstructures, with generated slices on top and corresponding segmented slices at bottom. All slices have Blaine values = 273. (b ~ c) have w/c = 0.35 and (d ~ e) have w/c = 0.50, covering the ages of 7, and 28 days; (f) is the legend of phases for the segmented microstructures.

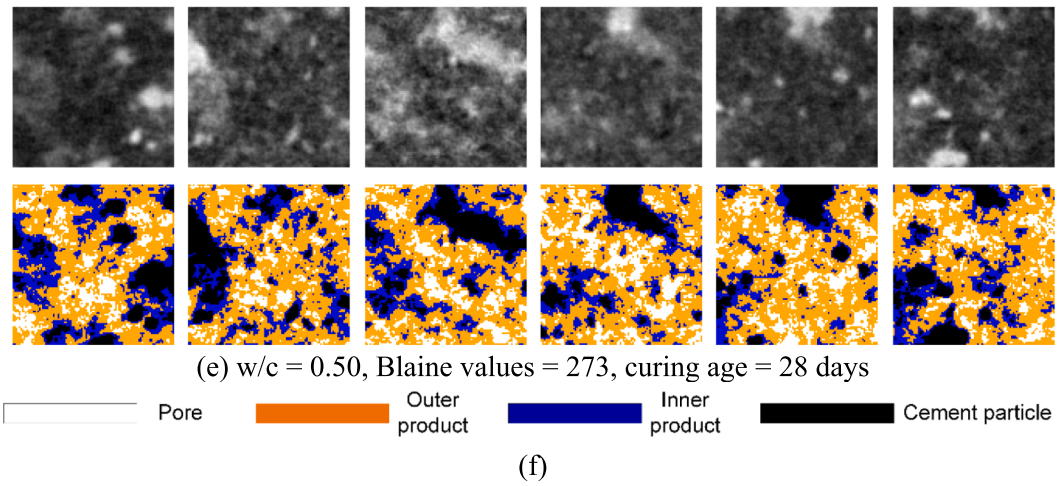


Fig. 9. (*continued*).

was set to 20, 18, 16, 14, 12, 10, and 10 μm for curing ages of 1 to 28 days, respectively. The watershed algorithm was subsequently applied to the negative distance map to partition the 3D microstructure into distinct particles. Each segmented particle was assigned a unique label, enabling further size-based statistical analysis. While this segmentation procedure involves manual tuning of the minimum distance threshold, the same values were consistently applied to both real and generated microstructures, ensuring fair and meaningful comparison. For more advanced and automated segmentation strategies, readers are referred to [75].

For each particle, its volume (V) was computed as the number of voxels in its labeled region, and the equivalent spherical diameter (D) was calculated using:

$$D = \left(\frac{6V}{\pi}\right)^{\frac{1}{3}} \tag{12}$$

The particle size distribution was further analyzed by calculating the cumulative volume distribution (CDF). For a given particle size D, the cumulative volume fraction F(D) was defined as:

$$F(D) = \frac{\sum_{i=1}^{n} V_i}{\sum_{i=1}^{N} V_i} \tag{13}$$

where V_i is the volume of the i-th particle, n represents particles with diameters less than or equal to D, and N is the total number of particles. CDF curves were generated for both real and generated datasets, enabling direct comparison of their particle size distributions.

5.3. Phase connectivity analysis by lineal path function

We characterize the connectivity of microstructural phases using the lineal path function (LPF) [21], a statistical descriptor developed to quantify the spatial continuity within a given phase. In this study, LPF is computed using the open-source PoreSpy library [76]. The LPF measures the probability that a line segment of a specified length lies entirely within a single phase, thus providing a quantitative assessment of phase connectivity. This approach has been effectively applied in previous studies to assess the statistical similarity of cementitious microstructures [33,44,77].

As demonstrated by micromechanical analyses [33,78,79], cracks are more likely to propagate through weaker phases such as outer product and pores. Therefore, these two phases play a dominant role in influencing the micromechanical behavior of the material. In this context, we compute the LPFs of both the pore and outer product phases for real and generated microstructures within the evaluation dataset. These connectivity statistics serve as indicators for evaluating the effectiveness of the proposed generative model in reproducing realistic microstructural features.

6. Results and discussion

6.1. Overview of generated microstructures

The TL-DiT is trained with a dataset of 3D microstructures of cement paste [13] and generated 28 types of microstructures, covering representative parameters including 2 w/c ratios (i.e., 0.35 and 0.50), 2 Blaine values (i.e., 273 and 391 m^2/kg^2), and 7 curing ages ranging from 1, 2, 3, 4, 7, 14, to 28 days. For each parameter set, 32 virtual microstructural cubes are generated and compared with 32 randomly selected cubes from the same categories of the real dataset.

Fig. 7 displays generated and real microstructures of 28 different parameter combinations. In each combination, the cube on the top is randomly selected from generated microstructures and the one on the bottom from real microstructures. The bright areas in the microstructure represent the unhydrated cement particles [14]. Beyond visual similarity, the impact of curing ages and Blaine values on the generated microstructures is evident. For different w/c ratios and Blaine values, a clear decrease in the amount of unhydrated cement particles along the axis of curing ages is observable, indicating that TL-DiT successfully captures the ongoing hydration process of cement particles. Additionally, microstructures with higher Blaine values exhibit smaller, more dispersed cement particles, confirming that TL-DiT captures the fineness of cement particles as represented by the Blaine values.

6.2. Statistics of generated microstructures

The histogram of grey values is a key statistical descriptor when analyzing microstructures represented by XCT or SEM images. These grey values are significant because they reflect the local density in XCT images and the atomic number in SEM images. Herein, the probability density function (PDF) and cumulative density function (CDF) averaged from 32 generated microstructures and 32 real microstructures of 28 different combinations of parameters are presented in Fig. 8. Despite noticeable shifts in grey value ranges across different material conditions, the model remains robust and accurately captures the underlying grey-level distributions. The coefficients of determination (R^2) between the PDFs and CDFs of generated and real microstructures were also calculated. The results illustrate that TL-DiT generates nearly identical

statistical distribution of grey values compared to the real dataset. All parameter combinations exhibited R^2 values greater than 0.995, indicating high similarity [14].

6.3. Microstructural phase assemblage and evolution

The cement paste is assumed as a four-phase composite including pore, outer product, inner product, and cement particle [80]. Based on the grey value statistics and an established hydration model, this paper calculates the threshold points S1, S2, and S3 for segmentation of the four-phase composites [63]. The threshold points of generated and real microstructures for 28 parameter combinations are shown in Fig. 9(a). The paired threshold points of the generated and real microstructures closely align with the 45-degree line, with a coefficient of determination of 0.996, highlighting the high correlation between the phase assemblage of real and generated microstructures. Fig. 9(b, c) and Fig. 9(d, e) present the 2D slices cropped every 19 μ m along the depth axis from segmented and real microstructural cubes, which are generated by the TL-DiT with Blaine values = 273 at curing ages of 7 and 28 days. The former has a w/c ratio of 0.35 while the latter has a w/c ratio of 0.50.

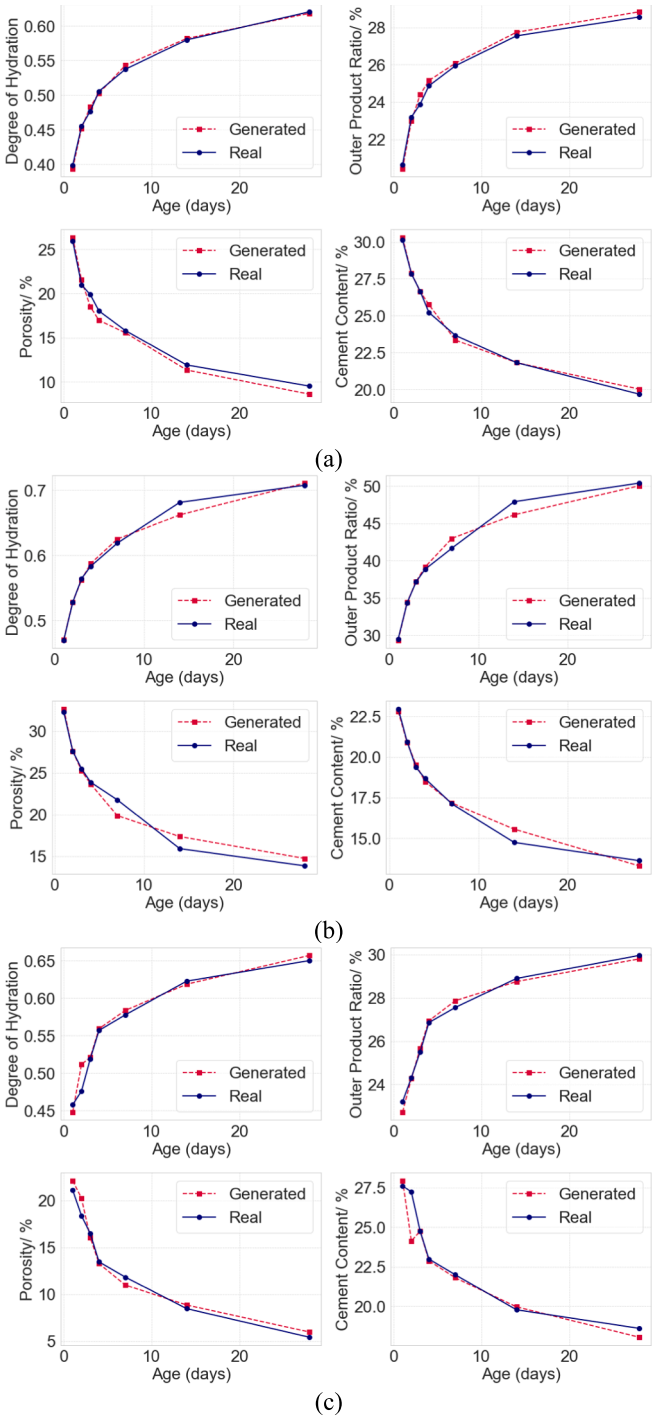


Fig. 10. Phase assemblage and evolution of 28 types of generated and real cubes: (a ~ d) represents the results of (w/c = 0.35, Blaine value = 273), (w/c = 0.50, Blaine value = 391), (w/c = 0.50, Blaine value = 273), (w/c = 0.50, Blaine value = 391) over 7 curing ages, respectively. For each w/c and Blaine value combination, the degree of hydration, outer product ratios, porosity, and cement content are presented.

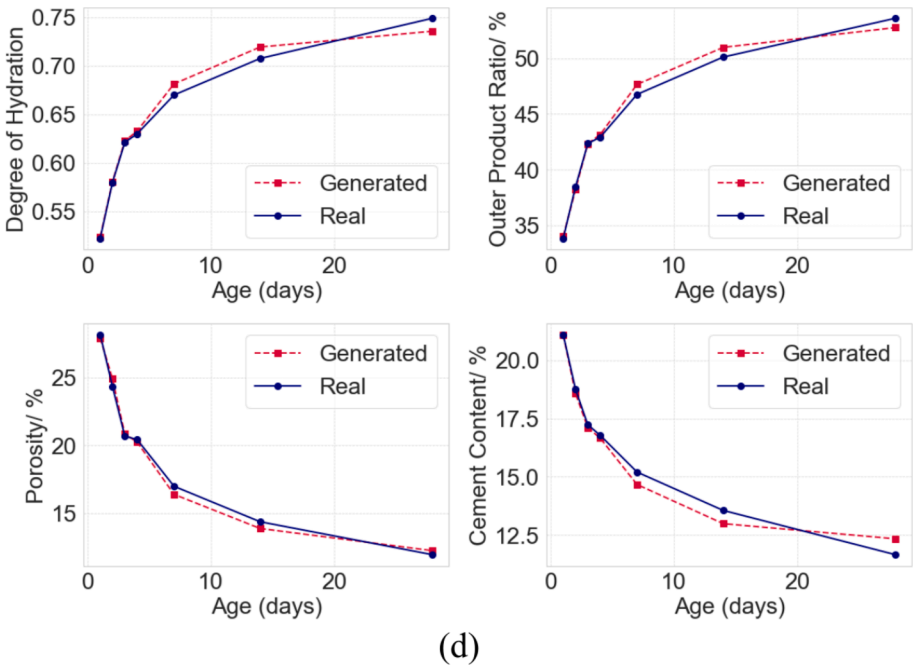


Fig. 10. (*continued*).

Although the distance between every two adjacent slices is relatively long (i.e., 19 μ m), the consistent change of microstructure along the depth can be seen with the existence of large cement particles, such as Fig. 9(c), suggesting the effectiveness of the depth transformer blocks in capturing the depth-wise microstructural information.

If comparing the microstructures at different ages (within Fig. 9(b, c) or Fig. 9(d, e)), it clearly illustrates that as the hydration reaction of cement continues, the porosity decreases and the volume of outer product and inner product increases, indicating a pore filling process by generation of hydration products. More interestingly, in the segmented results at a more mature age (Fig. 9(c, e)), the cement particles are generally surrounded by an inner product layer, which is in accordance with common understandings of microstructures of cement paste [81]: As the cement particle dissolves and reacts, it forms a dense layer of hydration products inner product around the particle. Comparing the microstructures with different w/c ratios, the hydration products in microstructure with low w/c ratio are dominated by inner product (Fig. 9(b, c)). On the other hand, the hydration products in microstructures with high w/c ratio are mainly comprised of outer product (Fig. 9(d, e)). The inner product either forms a clear rim structure around the cement particles or sparsely distributes in the outer product matrix. The sparsely-distributed inner product is likely to be the Calcium Hydroxide (CH) which is often produced in the water-filled space and mixed with the outer product [14].

Furthermore, all the threshold points in Fig. 9(a) are used to segment 32 generated and real microstructure cubes of each parameter combination. The volume fractions of each microstructural phase are then calculated. The hydration degree, outer product ratio, porosity, and cement content of all parameter combinations are shown in Fig. 10. The results show strong agreement in evolution of volume fraction of each phase between generated and real microstructures. Moreover, such results tell an evolution process depicted by classical hydration theories [80]: as the curing ages increases, the hydration reaction continues to consume cement, produces hydration products that fill the pores. Besides, the effects of w/c ratios and Blaine values on the microstructures are also clearly illustrated, which are in good agreement with common understandings of cementitious materials [82], listed as below:

- Comparing Fig. 10(a) with Fig. 10(b), or Fig. 10(c) with Fig. 10(d), it is found that higher w/c ratio generally leads to higher degree of hydration but creates more pores. Moreover, microstructure with high w/c ratios usually generates more outer product than the lower ones [72].
- Comparing Fig. 10(a) with Fig. 10(c), or Fig. 10(b) with Fig. 10(d), it is found that cement with higher Blaine values reacts faster, causing faster increase in hydration degrees that consumes more cement, produces more hydration products, and therefore less pores are created.

In summary, both the qualitative and quantitative microstructural analysis above prove that the TL-DiT captures the microstructural evolution of cement paste inducted by hydration reaction.

6.4. *Size distribution of cement particles*

Based on the segmented microstructure, the cement particles are separated by SciPy library's watershed segmentation algorithm [56] and calculated assuming a spherical geometry. Fig. 11 presents the accumulative volume distribution (AVD) of cement particles in microstructures with Blaine value = 273 and w/c = 0.35 at the ages of 1, 7, and 28 days, which is calculated by averaging the results of 32 microstructural cubes. The segmented cement particles of real and generated microstructures are presented in the middle and right of Fig. 11, respectively. Except for the visual similarity of the cement particles, a decreasing particle count is noted with increasing curing ages, indicating ongoing consumption of cement during the hydration reaction. Generally, the AVD curves of cement particles of both real and generated microstructures show high resemblance, underscoring the effectiveness of TL-DiT in capturing the morphology of cement particles in the microstructure. Across different ages, the AVD curves exhibit a nearly self-similar distribution shape. However, because of the cement consumed within the first 7 days, a clear decrease in cement particle sizes can be observed from the age of 1 day to 7 and 28 days, by comparing Fig. 11(a) with Fig. 11(b, c).

Nevertheless, when it comes to large particle sizes (>50 μm), the

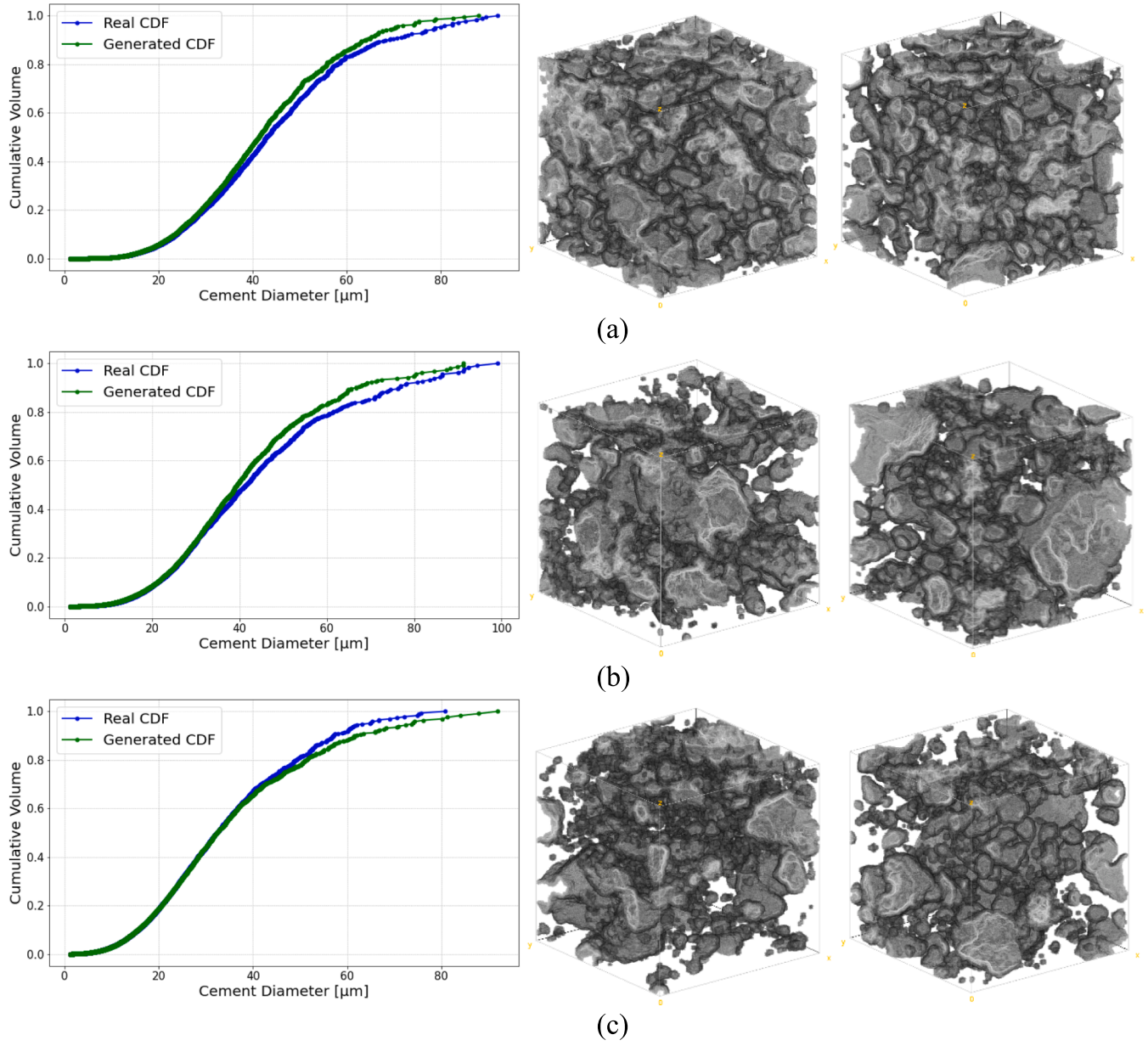


Fig. 11. Size distribution of cement particles with the parameter set of Blaine value = 273 and w/c = 0.35: (a ~ c) represents the results of 1, 3, and 7 days. The left is the pore size distribution curve, middle is the cement particles of real microstructures, right is the cement particles of generated microstructures.

AVD curves in Fig. 11 suggest inconsistency between the generated and real microstructures. Such inconsistency may be attributed to the "wall effect" introduced by the microstructural cube's finite boundaries. The large particles are more susceptible to being intercepted by the cube boundaries and therefore the morphology information of cement particles is obscured. Such boundary effects are more evident when comparing characteristic particle sizes. The AVD curves for all parameters were calculated, and characteristic diameters—D_{10}, D_{50}, and D_{90}—were extracted to represent particle diameters at the 10th, 50th, and 90th percentiles of accumulative volume, as depicted in Fig. 12.

Overall, there is good agreement between the real and generated microstructures, especially with small and medium particle sizes D_{10} and D_{50}. The particle size distribution in the generated microstructures behaves similarly to the real ones, with particle sizes predominantly decreasing over the initial 7 days and stabilizing afterward. This pattern aligns with the phase evolution trends observed in the previous section. However, a noticeable discrepancy in the D_{90} evolution is present between generated and real microstructures, possibly due to the "wall effect". Such wall effects significantly influence the morphology representation of the large cement particles. Clear evidence is that the D_{90} in real microstructures evolves in a disordering and non-continuous way, indicating that the morphology of intercepted cement particles may be more difficult to capture.

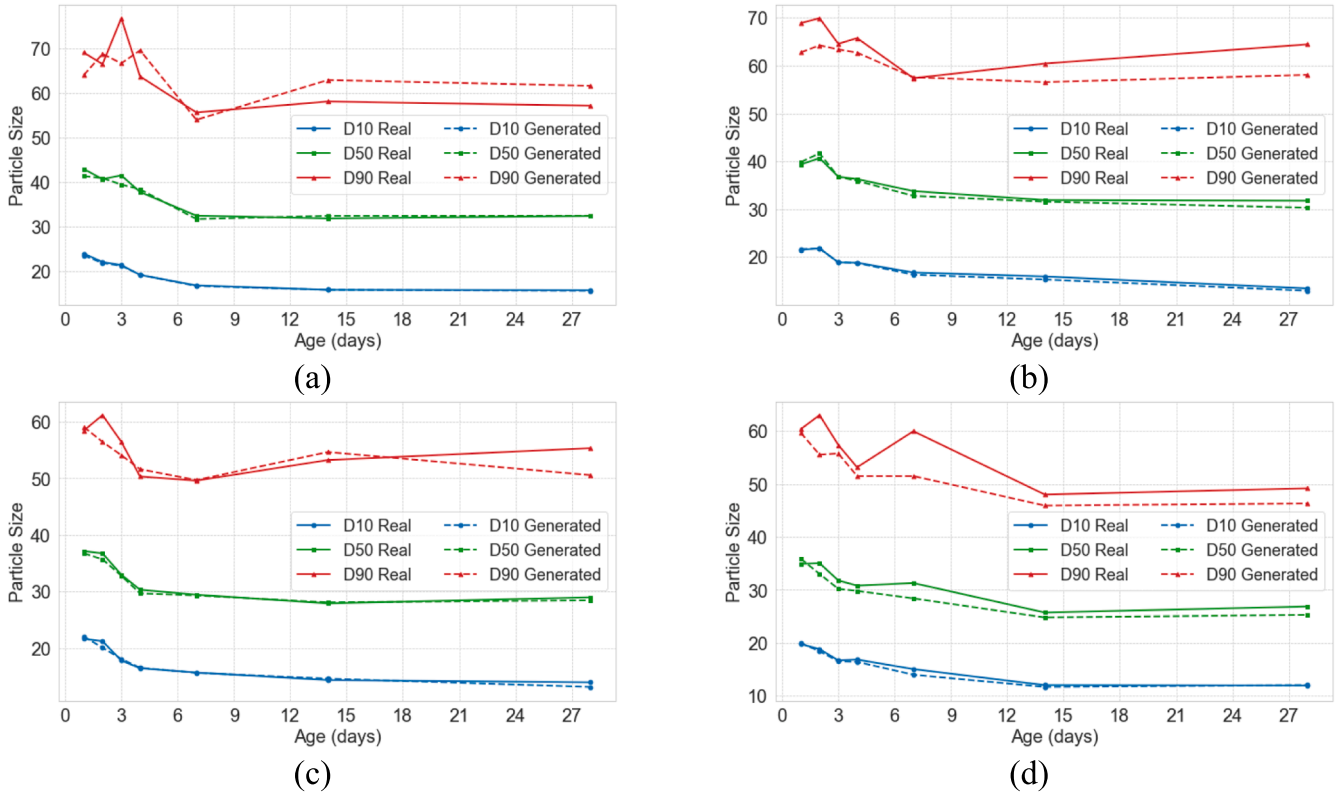


Fig. 12. Evolution of characteristic sizes of cement particle over the curing ages from 1 to 28 days: (a ~ d) represents results of (w/c = 0.35, Blaine value = 273), (w/c = 0.50, Blaine value = 391), (w/c = 0.50, Blaine value = 273), (w/c = 0.50, Blaine value = 391).

6.5. Phase connectivity

Using the LPF, the phase connectivity of the pore and outer product phases is quantified. Fig. 13 presents the results for microstructures with two representative mixtures—namely, w/c = 0.35 with a Blaine value of 273, and w/c = 0.50 with a Blaine value of 391—at two curing ages (7 and 28 days). Overall, the LPF results indicate a strong similarity in the spatial connectivity of both the pore and outer product phases between the generated and real microstructures.

In general, the outer product phase exhibits higher LPF values over a broader range of path lengths compared to the pore phase. However, an exception is observed in the 7-day microstructure produced with coarser cement and a lower w/c ratio (Fig. 13(a)), where the pore phase shows higher connectivity than the outer product. Such discrepancy can be explained by mainly two reasons:

- The combination of coarser cement and lower w/c ratio results in a slower hydration process. This is evident when comparing the degree of hydration between the samples in Fig. 13(a) and Fig. 13(c)—0.54 versus 0.67, respectively, as discussed in Section 6.2. As hydration progresses to 28 days, a marked reduction in pore connectivity is observed (Fig. 13(b)), which aligns with expectations based on hydration kinetics.
- A lower w/c ratio generally leads to a reduction in the ratio of outer product to inner product, indicating that outer product may not be the dominant hydration product in the corresponding sample. This pattern is consistent with the theoretical relationship described by the adopted hydration model (i.e., Eq. (11)) and well demonstrated in Sections 6.2 and 6.3.

6.6. Confusion matrix of microstructures given by the classification model

For the two-stage latent representation (Fig. 1(a)), VGG-16 classification model is trained to provide the perceptual losses for training the VQVAE. The VGG-16 model in the stage I was trained to classify the Blaine values, w/c ratios, and curing ages of 2D slices of real microstructures. Herein, this model is used to further validate the TL-DiT by comparing the difference between generated and real microstructures.

From each of the 28 parameter combinations, 128 2D slices are randomly cropped from 32 real and generated microstructural cubes. Thus, 3584 2D slices of real and generated microstructures are sampled, which can be used as the input testing data for the VGG-16 classification model. Fig. 14 presents the confusion matrix derived by the testing data, with the left column being the results of real microstructures and right being the generated microstructures. The results show that the VGG-16 model obtains excellent prediction accuracy on the dataset of both real and generated microstructures. The performance on the real microstructures is slightly better than that on generated microstructures, which is not surprising because the real microstructures are the source of training set for the VGG-16 model.

Moreover, the detailed performance of the VGG-16 on the two datasets shows similar characteristics. On the real microstructure dataset, the VGG-16 gets relatively low accuracy (i.e., 94 %) when it tries to classify the microstructures at the curing age of 14 days, with a low

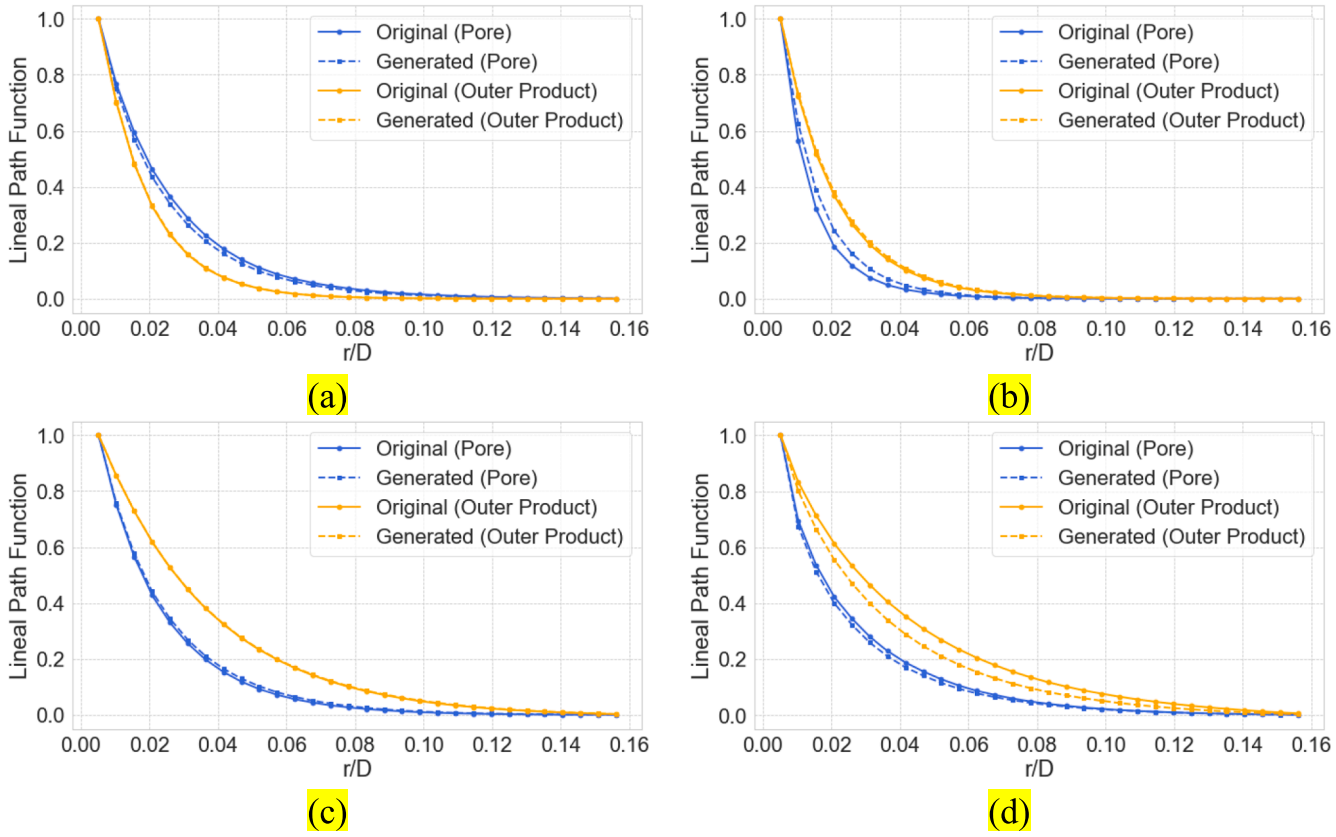


Fig. 13. LPF function calculated for pore and outer product: (a) w/c = 0.35, Blaine value = 273, age = 7 days; (b) w/c = 0.35, Blaine value = 273, age = 28 days; (c) w/c = 0.50, Blaine value = 391, age = 7 days; (d) w/c = 0.50, Blaine value = 391, age = 28 days. (Note: r is the line length and D is the side length of each microstructure).

probability to classify the 14-day microstructures as 4, 7, or 28-day ones. The reason for this is that over 50 % of the hydration reaction happens in the first 3 days, and afterwards the microstructures evolve much slower, as also shown in the analysis of phase assemblage evolution and cement particle size. Therefore, the microstructures after 14 days are more similar and more difficult to classify. Another potential reason may lie in the difference in XCT imaging settings. Most interestingly, the same pattern is also found in the results of the generated microstructure dataset. The prediction of the VGG-16 over the 14-day generated microstructures obtains relatively low accuracy (i.e., 91 %), with a low probability to be falsely classified as 4, 7, or 28-day microstructures. Such similarity holds also for the prediction of 7-day microstructures. Overall, the testing results strongly suggest the resemblance between the real and generated microstructures, underscoring the effectiveness of the TL-DiT in generating high-fidelity microstructures of cement paste.

7. Conclusions

Encoding cement paste microstructures into a two-stage latent space using VQVAEs yields highly compact latent representations while preserving essential microstructural information. The diffusion transformer, enhanced with slice-depth blocks, effectively captures the latent cube distributions, enabling rapid sampling and the generation of realistic microstructures.

The generated microstructures exhibit a statistical distribution of XCT grey values that closely matches that of real microstructures. This alignment indicates a high level of consistency in micromechanical characteristics such as elastic modulus and hardness, which have been shown in the authors' previous studies [79,83,84] to correlate with grey values. Additionally, strong agreement is observed in the evolution of microstructural phase assemblage. Predictions obtained using the VGG-16 classification model for both real and generated microstructures show excellent accuracy, with only minor discrepancies in a small number of false predictions.

Cement particle size distributions in the generated and real microstructures exhibit a similar pattern, particularly in the distribution of small and medium-sized particles. However, discrepancies are noted in the large-particle range, where both generated and real microstructures display a disordered and discontinuous evolution pattern. This deviation is primarily attributed to wall effects, where large particles are intercepted by cube boundaries, resulting in the loss of morphological information. Such limitations stem from the restricted size of the representative volume element (RVE) in the real dataset. Although increasing the RVE size can mitigate wall effects, this comes at a significant computational cost, and boundary-induced artifacts remain unavoidable. A promising solution lies in the implementation of continuity conditions across adjacent microstructural cubes. By conditioning the generation process on neighboring boundaries, the model could preserve morphological coherence across cube edges. This would enable the generation of arbitrarily large and seamless microstructures through tiling or autoregressive stitching of smaller volumes.

Another key limitation lies in the current conditioning mechanism, which treats the input variables—curing age, w/c ratio, and Blaine fineness—as discrete classes, implemented via learnable embedding layers. While these embeddings exist in a continuous latent space, the model does not explicitly support continuous or interpolated inputs (e.

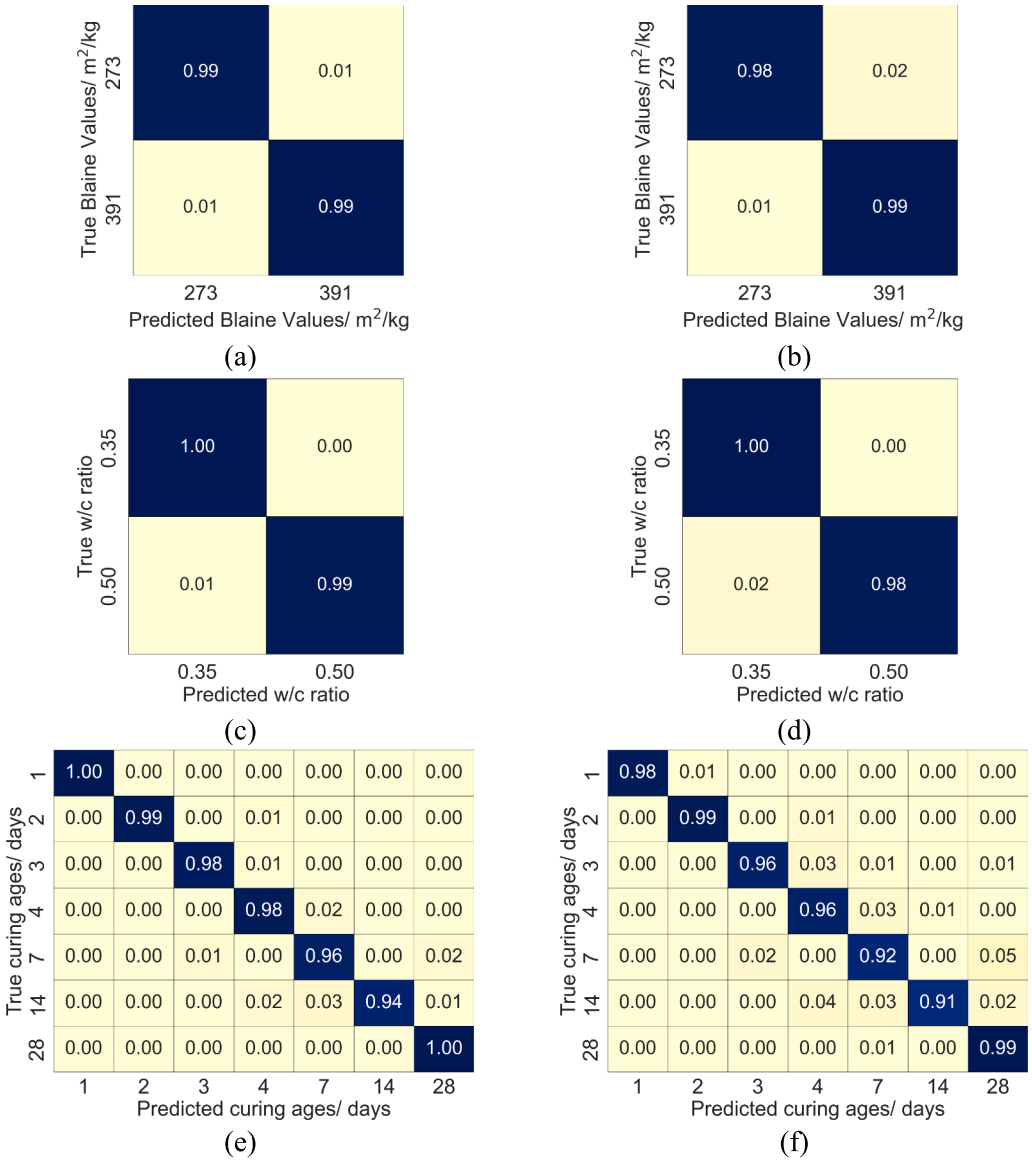


Fig. 14. Confusion matrix of testing results of the real and generated microstructures using the VGG-16 classification model: (a, c, e) and (b, d, f) are the confusion matrixes of Blaine values, w/c ratios, and curing ages of real and generated microstructures, respectively.

g., a w/c ratio of 0.42). Moreover, the current dataset is limited to OPC, where Blaine fineness remains a widely accepted parameter to characterize particle fineness. For alternative or blended cementitious systems, Blaine fineness may not be sufficient, and other influential parameters—such as chemical composition, curing temperature, and humidity—would need to be considered. These variables, however, are not available in the present dataset and therefore could not be incorporated into the model. Extending the conditioning scheme to handle real-valued inputs, for instance through an MLP-based encoder, and expanding the dataset to cover a broader range of material systems, would enable the generation of microstructures under more diverse or intermediate conditions. We consider this a meaningful direction for future work, particularly in the context of predictive microstructure generation across a wider class of cementitious materials.

The findings presented in this study highlight the potential of TL-DiT as an efficient tool for generating high-fidelity microstructures of Portland cement paste. This deep learning-based approach demonstrates that generative AI can serve as a cost-effective alternative to conventional imaging techniques, such as SEM and XCT, in capturing the microstructure of ordinary Portland cement paste. By providing high-fidelity microstructural data in a computationally efficient manner, TL-DiT offers new possibilities for multiscale investigations into the microstructure-property relationships of concrete materials and structures.

CRediT authorship contribution statement

Minfei Liang: Writing – review & editing, Writing – original draft, Visualization, Software, Investigation, Conceptualization. **Kun Feng:** Writing – review & editing, Supervision, Funding acquisition. **Jinbao Xie:** Writing – review & editing, Investigation. **Yuyang Wei:** Writing – review & editing, Visualization. **Sonia Contera:** Writing – review & editing, Supervision. **Erik Schlangen:** Writing – review & editing, Supervision, Funding acquisition. **Branko Šavija:** Writing – review & editing, Supervision, Investigation, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

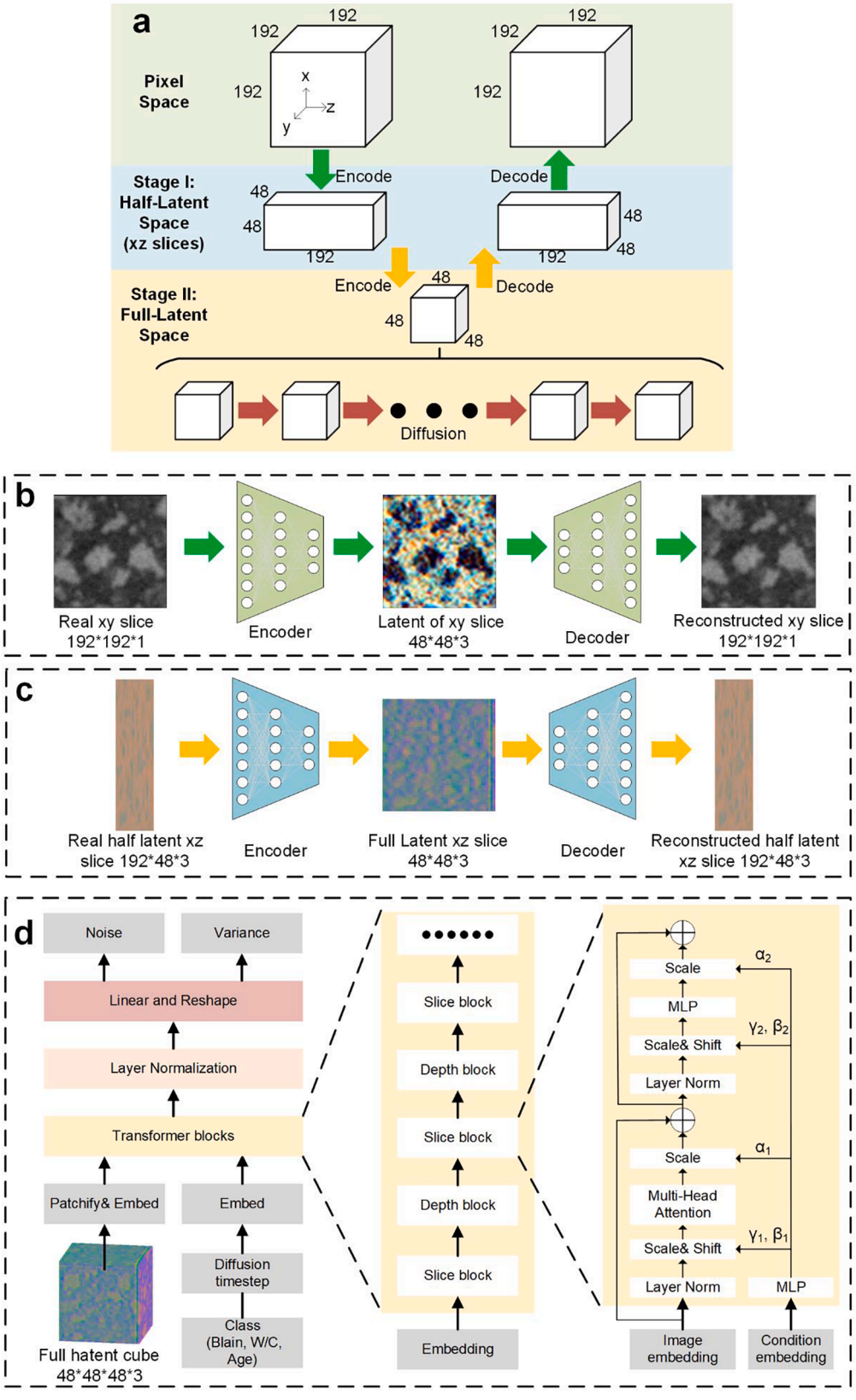
Minfei Liang, Yuyang Wei, and Sonia Contera would like to acknowledge the funding support provided by the European Union within the Project 'ZEBAI - Innovative methodologies for the design of Zero-Emission and cost-effective Buildings enhanced by Artificial Intelligence' HORIZON-CL5-2023-D4-01-01 [grant agreement No: 101138678, DOI 10.3030/101138678]. Kun Feng acknowledges the financial support of the National Key R&D Program of China, Grant Number 2021YFB2600900, and the National Natural Science Foundation of China, Grant Number 52378418. Branko Šavija acknowledges the financial support of the European Research Council (ERC) within the framework of the ERC Starting Grant Project "Auxetic Cementitious Composites by 3D printing (ACC-3D)", Grant Agreement Number 101041342.

Appendix A. Overall architecture of the TL-DiT

The overall model architecture of the Two-stage Diffusion Transformer (TL-DiT) is shown in Fig. S1. Inspired by latent diffusion models [48], TL-DiT performs generative tasks within a latent space to significantly reduce the computational demands associated with processing the real 3D dataset. As in Fig. S1(a), stage I begins by encoding the xy slices of a 192 × 192 × 192 microstructure of cement pastes into a half latent space to obtain a 48 × 48 × 192 half latent representation, by stacking the encoded xy slices along the z axis. Then, in stage II, the model encodes the xz slices of the half latent into the full latent space to obtain a 48 × 48 × 48 full latent representation, by stacking the encoded xz slices along the y axis. Finally, the full latent representation is decoded in a sequential manner to reconstruct and generate the 3D microstructure of the cement paste.

This paper used two vector quantized variational autoencoders (VQVAE) [49] to bridge the pixel space, half latent space, and full latent space (Fig. S1(b, c)). The backbone of the VQVAE is a convolutional UNet [50] architecture, aiming to autoregressively learn the discrete latent representation and model their distributions. The VQVAE is usually trained by reconstruction loss, commitment loss, and codebook loss. More importantly, to maintain the major microstructural information in the data compression stages, this model incorporated the perceptual and adversarial loss to guide the training of VQVAE for reservation of essential and detailed microstructural information [51]. The perceptual loss is derived by a VGG-16 model pretrained on the microstructure data. The adversarial loss involves a convolutional neural network as the discriminator and the VQVAE as the generator.

In the full latent space, this paper used a diffusion transformer [53,54] to model the distribution of the 48 × 48 × 48 full latent representation (Fig. S1(d)). The full latent is patchified and embedded to form initial feature representations, which can then be processed by a series of transformer blocks. The transformer backbone comprises two transformer block types: slice transformer block and depth transformer block. Each block consists of a multi-head attention layer and a multi-layer perceptron (MLP) layer. The slice block captures the distribution of the microstructure over the slice, while the depth block focuses on the changes of the microstructure along the depth. When a full latent cube is patchified and embedded with positional tokens, it forms a token sequence with dimensions D × W × H × d, representing depth, width, height, and token dimension. For the slice block, tokens are reorganized into D × n × d (where n = W × H) to extract the slice-wise microstructural representation. For the depth block, tokens are rearranged into n × D × d for depth-wise analysis. Within each transformer block, the tokens of diffusion timesteps and class conditions (i.e., w/c, Blaine values, and curing ages) are first processed by an MLP to obtain six parameters, which can then be used for adaptive layer normalization [53–55], enabling conditional generation.



(caption on next page)

Fig. S1. Overview of the TL-DiT architecture: (a) Overall workflow. In stage I, the xy slices of 192 × 192 × 192 microstructure are encoded by the first VQVAE encoder to form a 48 × 48 × 192 half-latent. Then, in stage II, the xz slices of the 48 × 48 × 192 half-latent are encoded by the second VQVAE encoder to form the 48 × 48 × 48 full-latent. Finally, a diffusion transformer is used in the full-latent space for generative tasks. The generated full-latent can then be decoded sequentially by the two VQVAE decoders to generate microstructures with specific inputs. (b) The first VQVAE in stage I, which encodes/decodes the 2D xy slices of the real microstructures. (c) The second VQVAE in stage II, which encodes/decodes the 2D xz slices of the half-latent. (d) The diffusion transformer in the full latent space, which utilizes sequentially connected alternating slice and depth blocks to capture the 3D information of the 48*48*48 full latent. The diffusion transformer takes the input of w/c ratios, Blaine values, and curing ages to generate corresponding full latent, which can then be sequentially decoded to generate the microstructure of specific type of cement paste.

Appendix B. Two-stage latent representation learning by VQVAE

The VQVAEs are utilized to encode the microstructures and decode the latent, bridging the pixel and latent space. The VQVAE configuration in stage I is shown in Fig. S2. Note that the VQVAE in stage II is the same as that in stage I, except for only the dimension of input/output.

The training of VQVAE is guided by reconstruction loss, codebook loss, commitment loss, perceptual loss, and adversarial loss, as below:

$$L_{VAE} = L_{recon} + L_{code} + \lambda_{com} L_{com} + \lambda_{pe} L_{pe} + \lambda_{ad} L_{ad} \tag{S1}$$

where the total loss L_{VAE} is the sum of reconstruction loss L_{recon}, codebook loss L_{code}, commitment loss L_{com}, and weighted loss of perceptual loss L_{pe} and adversarial loss L_{ad}. λ_{com}, λ_{pe} and λ_{ad} represent the wights of commitment, perceptual and adversarial loss term, which are 0.2, 1 and 0.8 in this study.

The reconstruction loss is for direct comparison of the real and reconstructed images, quantified by L2 loss. The codebook loss is for learning the discrete embedding space (i.e., codebook) based on vector quantization algorithms. The commitment loss is for regularizing the grow of latent space to make the latent commit to the codebook. The reconstruction loss, codebook loss and commitment loss are as below [49]:

$$L_{recon} = \|x_o - x_r\|^2$$

$$L_{code} = \|sg(z) - e\|^2$$

$$L_{com} = \|sg(e) - z\|^2$$

where x_o, x_r, z, and e are real image, reconstructed image, latent, and codebook vectors, respectively. The sg(·) operator is the stopgradient operator. In addition, to reserve the image details, this model also added perceptual loss and adversarial loss terms, which will be introduced in the following subsections.

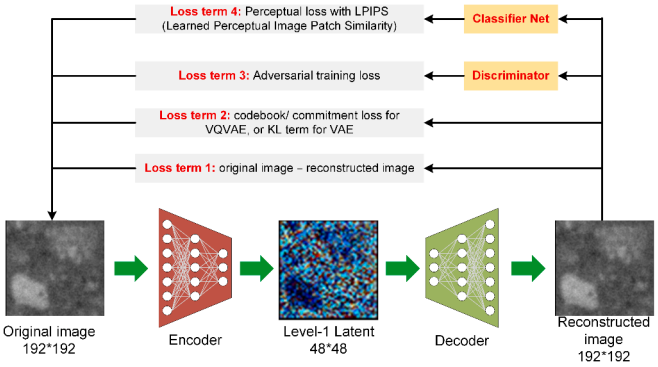


Fig. S2. VQVAE configuration in Stage I

B.1 U-net architecture

This paper constructed a U-Net [50] as the backbone of the VQVAE for the two-stage latent representation learning. The U-Net architecture is shown in Fig. S3, with the downward path being the encoder and the upward path being the decoder. Both encoder and decoder use residual convolutional blocks (i.e., ResNet Conv block in Fig. S3) and multihead attention layers to extract representations while progressively increasing the channel numbers. The ResNet Conv block mainly comprises residual convolutional layer [85], group normalization layer [86], and SiLU activation layer (Sigmoid-Weighted Linear Units) [87]. Downsample and Upsample blocks are used in encoder and decoder respectively, which decreases/increases the size of images by a factor of 2 using a convolutional layer with a stride of 2. Connecting the downward and upward block is a codebook embedding layer, where the continuous results are embedded into discrete latent space, forming the latent representation in stage I and stage II. The numbers of parameters of the VQVAEs in stage I and stage II are 9,922,844 and 9355678.

This study specified the channel size of latent as 48 × 48 × 3 and the codebook size of 8192. Although it is possible to further decrease the size of latent and codebook, which may benefit the computational efficiency, this work found that the current settings best fulfill the objective of high-fidelity reconstruction of the microstructures, which greatly benefit the latent diffusion task. Note that both Stage I and Stage II use almost the same settings of VQVAE, except for that the VQVAE in stage II has a different input size of 192 × 48 × 3. In this case, the upsample/downsample blocks only work on the first dimension, leading to the desired full-latent size 48 × 48 × 3.

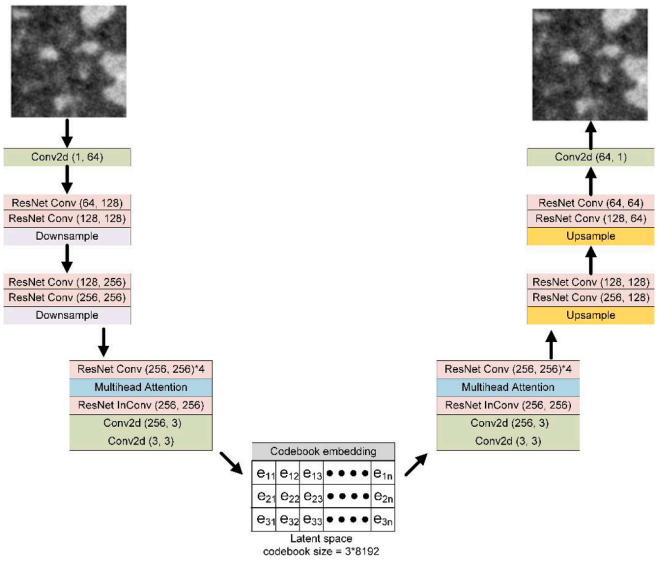


Fig. S3. UNet architecture for VQVAE

B.2 Perceptual loss

This work employs the Learned Perceptual Image Patch Similarity (LPIPS) [57] metric to compute the perceptual loss between generated and real microstructures, focusing on high-level feature similarity rather than pixel-by-pixel differences. Perceptual loss is advantageous for image generation tasks, as it captures semantic similarities and high-level structural information, which are essential for accurately reproducing microstructural details in cement paste images.

This work uses a VGG-16 [88] convolutional network to extract the high-level feature of microstructures. To ensure such extraction capabilities, the VGG-16 is first trained with a classification task, using the real microstructure dataset along with corresponding class labels (Fig. S4(a)). Then, the pretrained VGG-16 model is used to extract the latent representation and calculate the perceptual loss based on the LPIPS method (Fig. S4(b)).

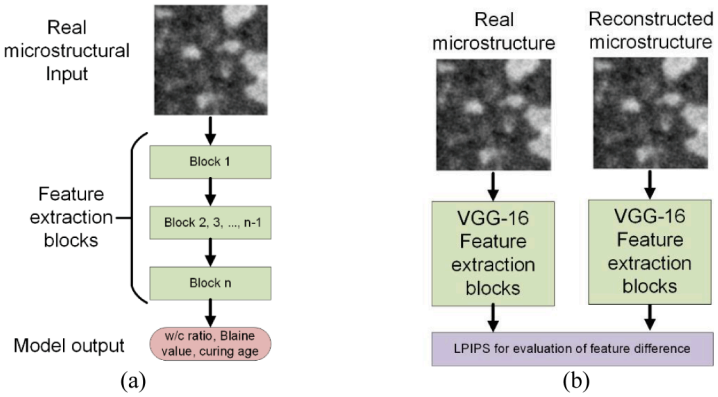


Fig. S4. Calculation of perceptual loss: VGG-16 classification network is first trained using the real microstructure and corresponding class labels (a). Then, the pretrained VGG-16 is used to extract features of both real and generated microstructures and then calculate the LPIPS loss (b).

Given a microstructure slice as input, the VGG-16 classification model aims to predict the w/c ratio, Blaine value, and curing age. The VGG-16 model is trained with the sum of cross entropy loss over the three kinds of labels, as below:

$$L_{vgg} = -\sum_{c_1=1}^{m_1} y_{o_1,c_1}\log(p_{o_1,c_1}) - \sum_{c_2=1}^{m_2} y_{o_2,c_2}\log(p_{o_2,c_2}) - \sum_{c_3=1}^{m_3} y_{o_3,c_3}\log(p_{o_3,c_3}) \tag{S3}$$

where m_1, m_2, m_3 are number of classes in w/c ratios, Blaine values, and curing ages respectively. $y_{o,c}$ is a binary indicator (0 or 1) if class label c is the correct classification for observation o. $p_{o,c}$ is the predicted probability of observation o is of class c.

This work then uses 81,016 microstructural slices to train the VGG-16 in stage I and stage II, with a ratio between training and validation set of 85:15. The loss entropy and prediction accuracy over each class labels of the VGG-16 model in stage I and stage II are shown in Fig. S5. This work stops the training if the average prediction accuracy on the validation set does not improve in 10 epochs, and therefore stopped training at 40 epochs. The training history shows that both VGG-16 models achieve accuracy close to 100 % at the end. The accuracy of VGG-16 in stage II is slightly lower than the stage I, due to potential information loss in the latent representation learning stage I. This paper did not conduct test over an extra testing set because of the large dataset size. However, the test of the VGG-16 classification model in stage I can be found in the main text in Fig. 12, showing high accuracy of the pretrained VGG-16 model.

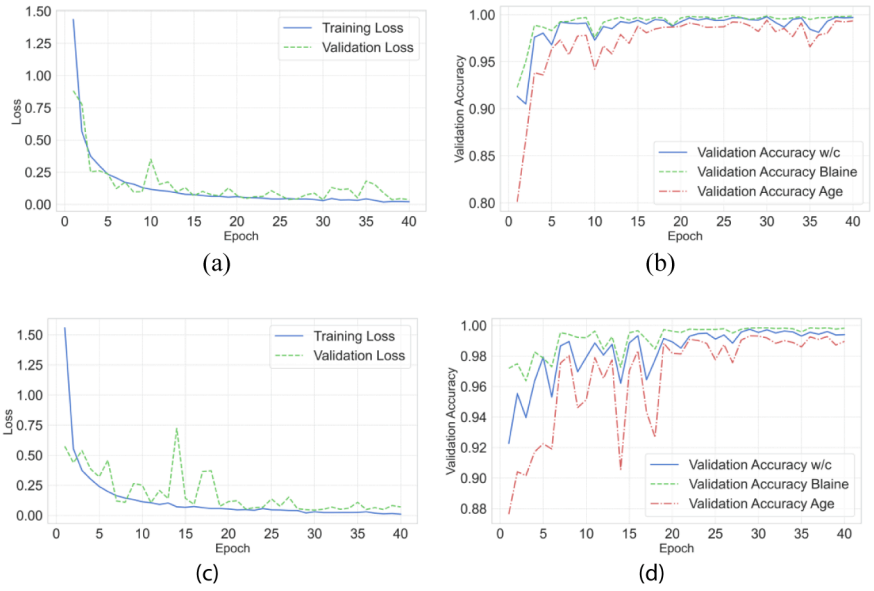


Fig. S5. Training history of VGG 16 classification network: (a ~ b) loss entropy and prediction accuracy of the VGG-16 in Stage I; (c ~ d) loss entropy and prediction accuracy of the VGG-16 in Stage II;

The trained VGG model is then used to calculate the perceptual loss following the LPIPS method (Fig. S4(b)), as below:

$$L_{pe}{}^{l} = \frac{1}{N_l}\sum_{i=1}^{N_l} \| w_l \left(\varphi_l(x_o)_i - \varphi_l(x_r)_i\right) \|^2 \tag{S4-1}$$

$$L_{pe} = \sum_{l} L_{pe}{}^{l} \tag{S4-2}$$

where N_l is the number of elements in the feature map at layer l. w_l represents learnable weights for each channel, applied via a 1x1 convolutional layer. These weights allow the model to learn channel-specific importance based on perceptual similarity. The total perceptual loss is obtained by summing the contributions across multiple layers.

B.3 Adversarial loss

To reserve the microstructural details, this model incorporates the adversarial loss in the training of the VQVAEs in both Stage I and Stage II. This adversarial loss calculation follows the min–max framework of Generative Adversarial Networks (GANs) [27], wherein a generator (VQVAE) and a discriminator are trained in a competitive setting. The discriminator is a 6-layer convolutional network, designed to distinguish between real and generated (fake) microstructural images, while the VQVAE serves as the generator, aiming to produce realistic microstructural reconstructions.

The discriminator architecture is designed to progressively capture fine-grained details of the microstructure. Each of the first five layers applies a convolution with a kernel size of 4 and a stride of 2, progressively increasing the feature channels from 1 to 32, 64, 128, 256, and 512. Batch normalization and leaky ReLU activation are employed in these layers to stabilize training and improve gradient flow. Following the PatchGAN architecture [59], the discriminator's final layer uses a sigmoid activation to produce a 6×6 feature map, where each cell represents the probability of a local patch being real or fake. This patch-based approach allows the discriminator to focus on local structure and texture, enhancing its ability to evaluate fine-grained details in microstructural images. Binary cross entropy is used to calculate the loss of discriminator, as below:

$$L_{disc} = -\frac{1}{n}\sum_{i=1}^{n} [\log D(x_i) + \log(1 - D(G(z_i)))] \tag{S5}$$

where the $D(\cdot)$ is the discriminator and $G(\cdot)$ is the generator. x_i and z_i are microstructural slices and latent fed to the decoder of the VQVAE respectively. The batch size is denoted by n. The Eq (S5) encourages the discriminator to assign a probability of 1 to real microstructures and 0 to generated (fake) samples. In contrary, the generator uses the loss function as below:

$$L_{gen} = \frac{1}{n}\sum_{i=1}^{n} \log(1 - D(G(z_i))) \tag{S6}$$

By minimizing L_{gen}, the generator is driven to produce synthetic samples that appear real to the discriminator, effectively maximizing the discriminator's error on fake samples. This adversarial training framework, when combined with the VQVAE model, facilitates the preservation of complex microstructural details, leading to more realistic reconstructions that retain high perceptual fidelity.

B.4 Training

In Stage I and Stage II, this paper trained the VQ-VAE models using a learning rate of 0.00005 and a batch size of 32 for both models. To mitigate instability in adversarial training, this work delayed the discriminator's initialization until 5,000 training steps, using a reduced learning rate of 0.000005 for the discriminator thereafter. Each stage was trained over 12 epochs on two Nvidia RTX 6000 GPUs (Ada architecture), each equipped with 48 GB of memory. The entire training process takes approximately 11 h.

The training history for both stages is depicted in in Fig. S5. In Stage I, the VQ-VAE training exhibits a marked instability at epoch 6, evidenced by a spike in the generator loss (Fig. S5(e)), which suggests that the generator may be misled by the discriminator. This instability affects other loss terms as well (Fig. S5(a, c)). In Stage II, the discriminator loss steadily decreases while the generator loss increases, reflecting a similarly imbalanced trend. Nevertheless, the reconstruction, codebook, and perceptual loss terms continue to decrease, indicating progressive training and stable convergence.

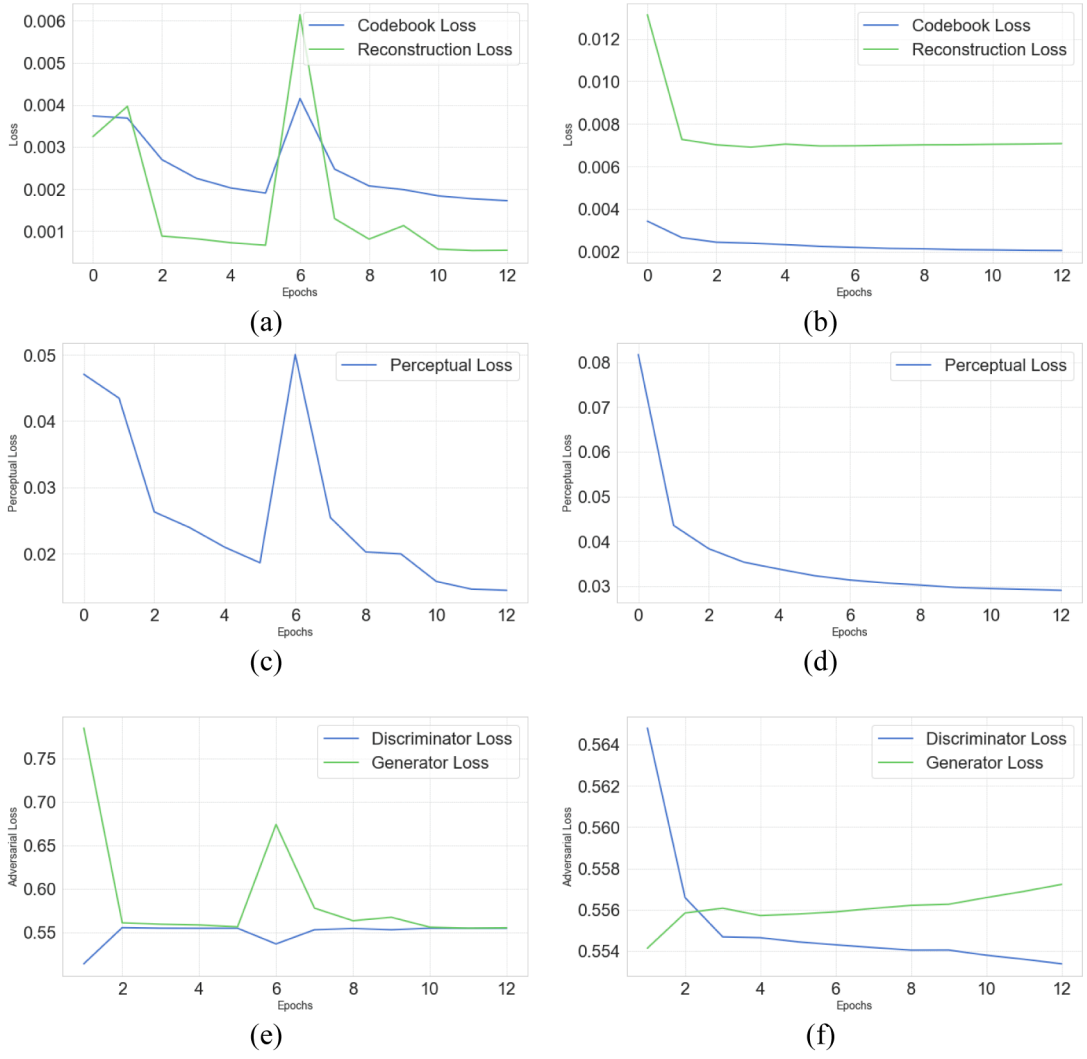


Fig. S6. Training history of reconstruction loss, codebook loss, perceptual loss, discriminator loss, and generator loss of VQVAEs: (a, c, e) loss history in Stage I; (b, d, e) loss history in Stage II.

In every epoch, we randomly checked the reconstructed and real microstructures. The comparison between randomly selected microstructures at 12th epochs are shown in Fig. S7. The reconstructed microstructures almost visually resemble the real microstructures. Furthermore, we checked the grey value histograms, latents, reconstructed and real microstructures of the stage I VQVAE, as shown in Fig. S8. The results show that the VQVAEs can not only preserve the visual similarity but also the statistical patterns of the microstructures.

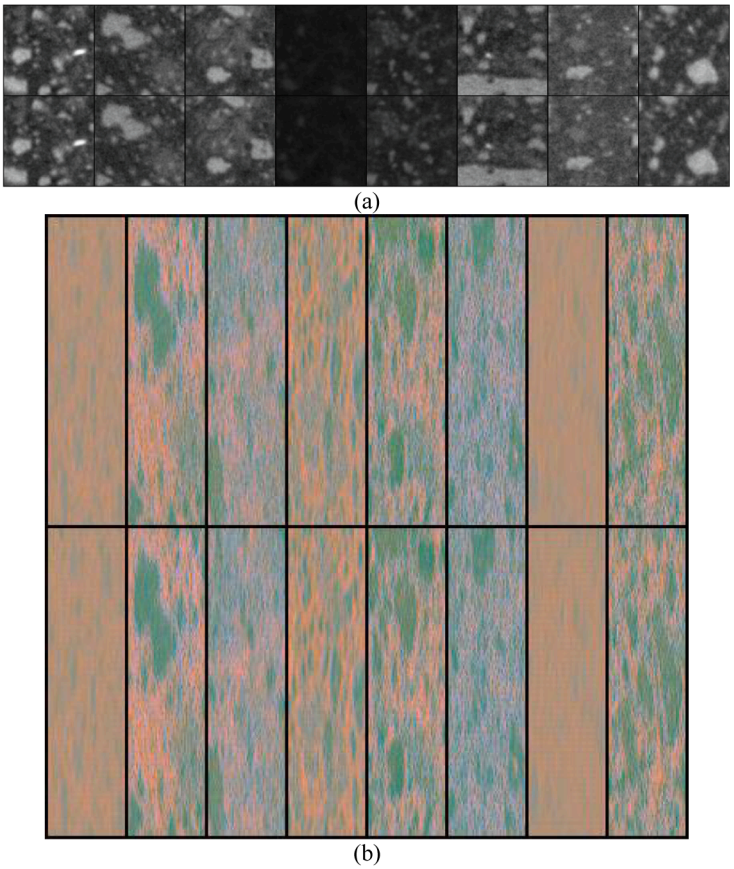


Fig. S7. Comparison of microstructure reconstruction in VQVAEs of stage I (a) and stage II (b) (Note: the microstructure on top is reconstructed and the one at bottom is real).

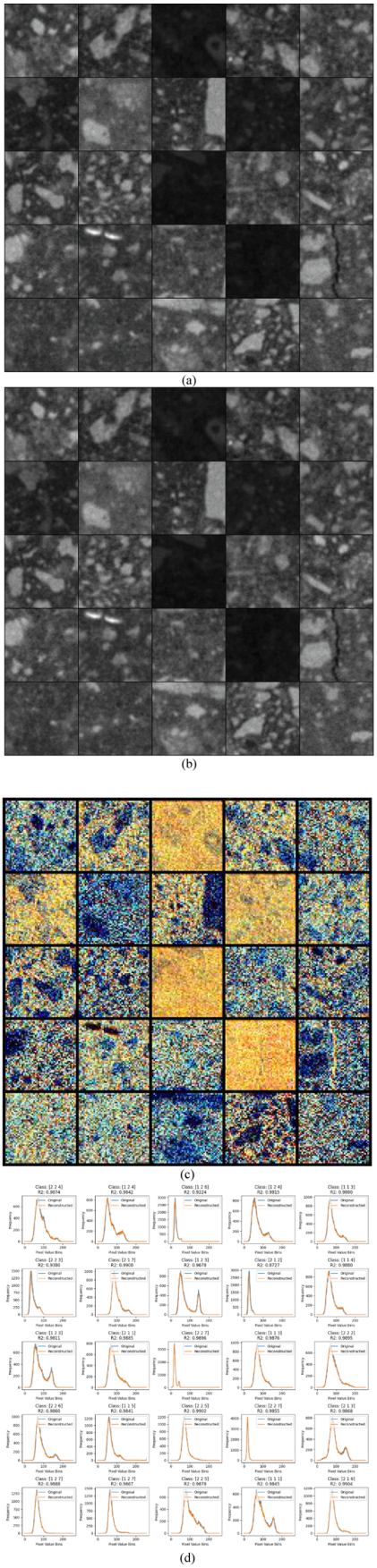


Fig. S8. 25 randomly selected microstructures from the VQVAE in Stage I: Reconstructed microstructures (a), real microstructures (b), latent (c), and histogram comparison (d).

Data availability

Data will be made available on request.

References

[1] G. Habert, S.A. Miller, V.M. John, J.L. Provis, A. Favier, A. Horvath, K.L. Scrivener, Environmental impacts and decarbonization strategies in the cement and concrete industries, Nat. Rev. Earth Environ. 1 (2020) 559–573, https://doi.org/10.1038/s43017-020-0093-3.

[2] S. Shirani, A. Cuesta, A. Morales-Cantero, I. Santacruz, A. Diaz, P. Trtik, M. Holler, A. Rack, B. Lukic, E. Brun, I.R. Salcedo, M.A.G. Aranda, 4D nanoimaging of early age cement hydration, Nat. Commun. 14 (2023) 2652, https://doi.org/10.1038/s41467-023-38380-1.

[3] D.P. Bentz, Modelling cement microstructure: Pixels, particles, and property prediction, Mater. Struct. 32 (1999) 187–195, https://doi.org/10.1007/BF02481514.

[4] V. Šmilauer, Z.K. Bittnar, Microstructure-based micromechanical prediction of elastic properties in hydrating cement paste, Cem. Concr. Res. 36 (2006) 1708–1718, https://doi.org/10.1016/j.cemconres.2006.05.014.

[5] A. Rhardane, F. Grondin, S.Y. Alam, Development of a micro-mechanical model for the determination of damage properties of cement pastes, Constr. Build. Mater. 261 (2020) 120514, https://doi.org/10.1016/j.conbuildmat.2020.120514.

[6] G. Constantinides, F.J. Ulm, The effect of two types of C-S-H on the elasticity of cement-based materials: results from nanoindentation and micromechanical modeling, Cem. Concr. Res. 34 (2004) 67–80, https://doi.org/10.1016/S0008-8846(03)00230-8.

[7] S. Liang, Y. Wei, Z. Wu, Multiscale modeling elastic properties of cement-based materials considering imperfect interface effect, Constr. Build. Mater. 154 (2017) 567–579, https://doi.org/10.1016/j.conbuildmat.2017.07.196.

[8] X. Gao, Y. Wei, W. Huang, Effect of individual phases on multiscale modeling mechanical properties of hardened cement paste, Constr. Build. Mater. 153 (2017) 25–35, https://doi.org/10.1016/j.conbuildmat.2017.07.074.

[9] E. Schlangen, E.J. Garboczi, Fracture simulations of concrete using lattice models: computational aspects, Eng. Fract. Mech. 57 (1997) 319–332, https://doi.org/10.1016/S0013-7944(97)00010-6.

[10] E. Schlangen, J.G.M. van Mier, Simple lattice model for numerical simulation of fracture of concrete materials and structures, Mater. Struct. 25 (1992) 534–542, https://doi.org/10.1007/BF02472449.

[11] Y. Gan, C. Romero Rodriguez, H. Zhang, E. Schlangen, K. van Breugel, B. Šavija, Modeling of microstructural effects on the creep of hardened cement paste using an experimentally informed lattice model, Comput. Aided Civ. Inf. Eng. 36 (2021) 560–576, https://doi.org/10.1111/mice.12659.

[12] H. Zhang, Y. Xu, Y. Gan, Z. Chang, E. Schlangen, B. Šavija, Microstructure informed micromechanical modelling of hydrated cement paste: techniques and challenges, Constr. Build. Mater. 251 (2020) 118983, https://doi.org/10.1016/J.CONBUILDMAT.2020.118983.

[13] M. Hlobil, I. Kumpová, A collection of three-dimensional datasets of hydrating cement paste, Data Brief 46 (2023) 108903, https://doi.org/10.1016/j.dib.2023.108903.

[14] K. Scrivener, R. Snellings, B. Lothenbach. A practical guide to microstructural analysis of cementitious materials, n.d.

[15] K. van Breugel, Numerical simulation of hydration and microstructural development in hardening cement-based materials (I) theory, Cem. Concr. Res. 25 (1995) 319–331, https://doi.org/10.1016/0008-8846(95)00017-8.

[16] K. van Breugel, Simulation of hydration and Formation of Structure in Hardening Cement-Based Materials, TU Delft, 1991.

[17] G. Ye, K. van Breugel, A.L.A. Fraaij, Three-dimensional microstructure analysis of numerically simulated cementitious materials, Cem. Concr. Res. 33 (2003) 215–222, https://doi.org/10.1016/S0008-8846(02)00889-X.

[18] S. Bishnoi, K.L. Scrivener, μic: a new platform for modelling the hydration of cements, Cem. Concr. Res. 39 (2009) 266–274, https://doi.org/10.1016/j.cemconres.2008.12.002.

[19] D.P. Bentz, Three-dimensional computer simulation of portland cement hydration and microstructure development, J. Am. Ceram. Soc. 80 (1997) 3–21, https://doi.org/10.1111/j.1151-2916.1997.tb02785.x.

[20] Z. Zhu, W. Xu, H. Chen, Z. Tan, Evolution of microstructures of cement paste via continuous-based hydration model of non-spherical cement particles, Compos. B Eng. 185 (2020) 107795, https://doi.org/10.1016/j.compositesb.2020.107795.

[21] S. Torquato, in: Random Heterogeneous Materials, Springer New York, New York, NY, 2002, https://doi.org/10.1007/978-1-4757-6355-3.

[22] S.-Y. Chung, T.-S. Han, S.-Y. Kim, T.-H. Lee, Investigation of the permeability of porous concrete reconstructed using probabilistic description methods, Constr. Build. Mater. 66 (2014) 760–770, https://doi.org/10.1016/j.conbuildmat.2014.06.013.

[23] S.-Y. Kim, J.-S. Kim, J.W. Kang, T.-S. Han, Construction of virtual interfacial transition zone (ITZ) samples of hydrated cement paste using extended stochastic optimization, Cem. Concr. Compos. 102 (2019) 84–93, https://doi.org/10.1016/j.cemconcomp.2019.04.012.

[24] M. Sahimi, P. Tahmasebi, Reconstruction, optimization, and design of heterogeneous materials and media: basic principles, computational algorithms, and applications, Phys. Rep. 939 (2021) 1–82, https://doi.org/10.1016/j.physrep.2021.09.003.

[25] M. Ragone, R. Shahabazian-Yassar, F. Mashayek, V. Yurkiv, Deep learning modeling in microscopy imaging: a review of materials science applications, Prog. Mater Sci. 138 (2023) 101165, https://doi.org/10.1016/j.pmatsci.2023.101165.

[26] D.P. Kingma, M. Welling. Auto-Encoding Variational Bayes, (2013).

[27] I.J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative Adversarial Networks, (2014).

[28] L. Zheng, K. Karapiperis, S. Kumar, D.M. Kochmann, Unifying the design space and optimizing linear and nonlinear truss metamaterials by generative modeling, Nat. Commun. 14 (2023) 7563, https://doi.org/10.1038/s41467-023-42068-x.

[29] Y. Kim, H.K. Park, J. Jung, P. Asghari-Rad, S. Lee, J.Y. Kim, H.G. Jung, H.S. Kim, Exploration of optimal microstructure and mechanical properties in continuous microstructure space using a variational autoencoder, Mater. Des. 202 (2021) 109544, https://doi.org/10.1016/j.matdes.2021.109544.

[30] M. Safiuddin, C.L. Reddy, G. Vasantada, C. Harsha, S. Gangolu, Establishing process-structure linkages using Generative Adversarial Networks, (2021).

[31] S. Chun, S. Roy, Y.T. Nguyen, J.B. Choi, H.S. Udaykumar, S.S. Baek, Deep learning for synthetic microstructure generation in a materials-by-design framework for heterogeneous energetic materials, Sci. Rep. 10 (2020) 13307, https://doi.org/10.1038/s41598-020-70149-0.

[32] B. Murgas, J. Stickel, S. Ghosh, Generative adversarial network (GAN) enabled statistically equivalent virtual microstructures (SEVM) for modeling cold spray formed bimodal polycrystals, NPJ Comput. Mater. 10 (2024) 32, https://doi.org/10.1038/s41524-024-01219-4.

[33] S.-W. Hong, S.-Y. Kim, K. Park, K. Terada, H. Lee, T.-S. Han, Mechanical property evaluation of 3D multi-phase cement paste microstructures reconstructed using generative adversarial networks, Cem. Concr. Compos. 152 (2024) 105646, https://doi.org/10.1016/j.cemconcomp.2024.105646.

[34] Simon J.D. Prince, Understanding Deep Learning, The MIT Press, 2023.

[35] M. Arjovsky, L. Bottou, Towards Principled Methods for Training Generative Adversarial Networks, (2017).

[36] [36] L. Metz, B. Poole, D. Pfau, J. Sohl-Dickstein, Unrolled Generative Adversarial Networks, (2016).

[37] J. Ho, A. Jain, P. Abbeel, Denoising Diffusion Probabilistic Models, (2020).

[38] X. Lyu, X. Ren, Microstructure reconstruction of 2D/3D random materials via diffusion-based deep generative models, Sci. Rep. 14 (2024) 5041, https://doi.org/10.1038/s41598-024-54861-9.

[39] A. Vahdat, J. Kautz, NVAE: A Deep Hierarchical Variational Autoencoder, (2020).

[40] P. Dhariwal, A. Nichol, Diffusion Models Beat GANs on Image Synthesis, (2021). https://doi.org/10.48550/ARXIV.2105.05233.

[41] E. Azqadan, H. Jahed, A. Arami, Predictive microstructure image generation using denoising diffusion probabilistic models, Acta Mater. 261 (2023) 119406, https://doi.org/10.1016/j.actamat.2023.119406.

[42] C. Düreth, P. Seibert, D. Rücker, S. Handford, M. Kästner, M. Gude, Conditional diffusion-based microstructure reconstruction, Mater. Today Commun. 35 (2023) 105608, https://doi.org/10.1016/j.mtcomm.2023.105608.

[43] K.-H. Lee, H.J. Lim, G.J. Yun, A data-driven framework for designing microstructure of multifunctional composites with deep-learned diffusion-based generative models, Eng. Appl. Artif. Intel. 129 (2024) 107590, https://doi.org/10.1016/j.engappai.2023.107590.

[44] M. Liang, K. Feng, S. He, Y. Gan, Y. Zhang, E. Schlangen, B. Šavija, Generation of cement paste microstructure using machine learning models, Dev. Built Environ. 21 (2025) 100624, https://doi.org/10.1016/j.dibe.2025.100624.

[45] K.-H. Lee, G.J. Yun, Multi-plane denoising diffusion-based dimensionality expansion for 2D-to-3D reconstruction of microstructures with harmonized sampling, NPJ Comput. Mater. 10 (2024) 99, https://doi.org/10.1038/s41524-024-01280-z.

[46] K.-H. Lee, G.J. Yun, Denoising diffusion-based synthetic generation of three-dimensional (3D) anisotropic microstructures from two-dimensional (2D) micrographs, Comput. Methods Appl. Mech. Eng. 423 (2024) 116876, https://doi.org/10.1016/j.cma.2024.116876.

[47] S. Kench, S.J. Cooper, Generating three-dimensional structures from a two-dimensional slice with generative adversarial network-based dimensionality expansion, Nat. Mach. Intell. 3 (2021) 299–305, https://doi.org/10.1038/s42256-021-00322-1.

[48] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, B. Ommer, High-Resolution Image Synthesis with Latent Diffusion Models, (2022). http://arxiv.org/abs/2112.10752 (accessed September 14, 2024).

[49] A. van den Oord, O. Vinyals, K. Kavukcuoglu, Neural Discrete Representation Learning, (2017). https://doi.org/10.48550/ARXIV.1711.00937.

[50] O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional Networks for Biomedical Image Segmentation, (2015).

[51] P. Esser, R. Rombach, B. Ommer, Taming Transformers for High-Resolution Image Synthesis, (2020). https://doi.org/10.48550/ARXIV.2012.09841.

[52] K. Simonyan, A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, (2014). https://doi.org/10.48550/ARXIV.1409.1556.

[53] W. Peebles, S. Xie, Scalable Diffusion Models with Transformers, (2022). https://doi.org/10.48550/ARXIV.2212.09748.

[54] X. Ma, Y. Wang, G. Jia, X. Chen, Z. Liu, Y.-F. Li, C. Chen, Y. Qiao, Latte: Latent Diffusion Transformer for Video Generation, (2024). https://doi.org/10.48550/ARXIV.2401.03048.

[55] E. Perez, F. Strub, H. de Vries, V. Dumoulin, A. Courville, FiLM: Visual Reasoning with a General Conditioning Layer, (2017). https://doi.org/10.48550/ARXIV.1709.07871.

[56] P. Virtanen, R. Gommers, T.E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S.J. van der Walt, M. Brett, J. Wilson, K.J. Millman, N. Mayorov, A.R.J. Nelson, E. Jones, R. Kern, E. Larson, C.

J. Carey, İ. Polat, Y. Feng, E.W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E.A. Quintero, C.R. Harris, A.M. Archibald, A.H. Ribeiro, F. Pedregosa, P. van Mulbregt, A. Vijaykumar, A.P. Bardelli, A. Rothberg, A. Hilboll, A. Kloeckner, A. Scopatz, A. Lee, A. Rokem, C.N. Woods, C. Fulton, C. Masson, C. Häggström, C. Fitzgerald, D.A. Nicholson, D.R. Hagen, D. V. Pasechnik, E. Olivetti, E. Martin, E. Wieser, F. Silva, F. Lenders, F. Wilhelm, G. Young, G.A. Price, G.-L. Ingold, G.E. Allen, G.R. Lee, H. Audren, I. Probst, J. P. Dietrich, J. Silterra, J.T. Webber, J. Slavič, J. Nothman, J. Buchner, J. Kulick, J. L. Schönberger, J.V. de Miranda Cardoso, J. Reimer, J. Harrington, J.L. C. Rodríguez, J. Nunez-Iglesias, J. Kuczynski, K. Tritz, M. Thoma, M. Newville, M. Kümmerer, M. Bolingbroke, M. Tartre, M. Pak, N.J. Smith, N. Nowaczyk, N. Shebanov, O. Pavlyk, P.A. Brodtkorb, P. Lee, R.T. McGibbon, R. Feldbauer, S. Lewis, S. Tygier, S. Sievert, S. Vigna, S. Peterson, S. More, T. Pudlik, T. Oshima, T.J. Pingel, T.P. Robitaille, T. Spura, T.R. Jones, T. Cera, T. Leslie, T. Zito, T. Krauss, U. Upadhyay, Y.O. Halchenko, Y. Vázquez-Baeza, SciPy 1.0: fundamental algorithms for scientific computing in Python, Nat. Methods 17 (2020) 261–272, https://doi.org/10.1038/s41592-019-0686-2.

[57] R. Zhang, P. Isola, A.A. Efros, E. Shechtman, O. Wang, The Unreasonable Effectiveness of Deep Features as a Perceptual Metric, (2018). https://doi.org/10.48550/ARXIV.1801.03924.

[58] K. Simonyan, A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, (2014). https://doi.org/10.48550/ARXIV.1409.1556.

[59] P. Isola, J.-Y. Zhu, T. Zhou, A.A. Efros, Image-to-Image Translation with Conditional Adversarial Networks, (2016). https://doi.org/10.48550/ARXIV.1611.07004.

[60] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016.

[61] D.P. Kingma, J. Ba, Adam: A Method for Stochastic Optimization, (2014).

[62] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, Improving Language Understanding by Generative Pre-Training, (n.d.).

[63] M. Hlobil, I. Kumpová, A. Hlobilová, Surface area and size distribution of cement particles in hydrating paste as indicators for the conceptualization of a cement paste representative volume element, Cem. Concr. Compos. 134 (2022) 104798, https://doi.org/10.1016/j.cemconcomp.2022.104798.

[64] M. Cervera, J. Oliver, T. Prato, Thermo-Chemo-Mechanical Model for Concrete. I: Hydration and Aging, Journal of Engineering Mechanics 125 (1999) 1018–1027. https://doi.org/10.1061/(ASCE)0733-9399(1999)125:9(1018).

[65] G. Di Luzio, G. Cusatis, Hygro-thermo-chemical modeling of high performance concrete. I: theory, Cem. Concr. Compos. 31 (2009) 301–308, https://doi.org/10.1016/J.CEMCONCOMP.2009.02.015.

[66] F.-J. Ulm, O. Coussy, Modeling of thermochemomechanical couplings of concrete at early ages, J. Eng. Mech. 121 (1995) 785–794, https://doi.org/10.1061/(ASCE)0733-9399(1995)121:7(785).

[67] G. Di Luzio, G. Cusatis, Hygro-thermo-chemical modeling of high-performance concrete. II: numerical implementation, calibration, and validation, Cem. Concr. Compos. 31 (2009) 309–324, https://doi.org/10.1016/J.CEMCONCOMP.2009.02.016.

[68] B. Klemczak, M. Batog, Z. Giergiczny, A. Żmij, Complex effect of concrete composition on the thermo-mechanical behaviour of mass concrete, Materials 11 (2018) 2207, https://doi.org/10.3390/ma11112207.

[69] D.P. Bentz, Influence of water-to-cement ratio on hydration kinetics: simple models based on spatial considerations, Cem. Concr. Res. 36 (2006) 238–244, https://doi.org/10.1016/j.cemconres.2005.04.014.

[70] K.A. Snyder, D.P. Bentz, Suspended hydration and loss of freezable water in cement pastes exposed to 90% relative humidity, Cem. Concr. Res. 34 (2004) 2045–2056, https://doi.org/10.1016/j.cemconres.2004.03.007.

[71] Studies of the Physical Properties of Hardened Portland Cement Paste, JP 43 (1946). https://doi.org/10.14359/15302.

[72] P.D. Tennis, H.M. Jennings, A model for two types of calcium silicate hydrate in the microstructure of Portland cement pastes, Cem. Concr. Res. 30 (2000) 855–863, https://doi.org/10.1016/S0008-8846(00)00257-X.

[73] M. Liang, Y. Gan, Z. Chang, Z. Wan, E. Schlangen, B. Šavija, Microstructure-informed deep convolutional neural network for predicting short-term creep modulus of cement paste, Cem. Concr. Res. 152 (2022) 106681, https://doi.org/10.1016/J.CEMCONRES.2021.106681.

[74] S. Van Der Walt, J.L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J.D. Warner, N. Yager, E. Gouillart, T. Yu, scikit-image: image processing in Python, PeerJ 2 (2014) e453.

[75] D.F.T. Razakamandimby R., H.-T. Kim, T.-S. Han, Y.H. Lee, K. Park, Recursive aggregate segmentation by erosion and reconstitution (RASER) to characterize concrete microstructure using complementarity of X-ray and neutron computed tomography, Cem. Concr. Compos. 148 (2024) 105437, https://doi.org/10.1016/j.cemconcomp.2024.105437.

[76] J. Gostick, Z. Khan, T. Tranter, M. Kok, M. Agnaou, M. Sadeghi, R. Jervis, PoreSpy: a python toolkit for quantitative analysis of porous media images, JOSS 4 (2019) 1296, https://doi.org/10.21105/joss.01296.

[77] T.-S. Han, X. Zhang, J.-S. Kim, S.-Y. Chung, J.-H. Lim, C. Linder, Area of lineal-path function for describing the pore microstructures of cement paste and their relations to the mechanical properties simulated from μ-CT microstructures, Cem. Concr. Compos. 89 (2018) 1–17, https://doi.org/10.1016/j.cemconcomp.2018.02.008.

[78] T.-S. Han, D. Eum, S.-Y. Kim, J.-S. Kim, J.-H. Lim, K. Park, D. Stephan, Multi-scale analysis framework for predicting tensile strength of cement paste by combining experiments and simulations, Cem. Concr. Compos. 139 (2023) 105006, https://doi.org/10.1016/j.cemconcomp.2023.105006.

[79] H. Zhang, B. Šavija, M. Luković, E. Schlangen, Experimentally informed micromechanical modelling of cement paste: an approach coupling X-ray computed tomography and statistical nanoindentation, Compos. B Eng. 157 (2019) 109–122, https://doi.org/10.1016/J.COMPOSITESB.2018.08.102.

[80] O. Bernard, F.-J. Ulm, E. Lemarchand, A multiscale micromechanics-hydration model for the early-age elastic properties of cement-based materials, Cem. Concr. Res. 33 (2003) 1293–1309, https://doi.org/10.1016/S0008-8846(03)00039-5.

[81] H.F.W. Taylor, in: Cement Chemistry, Thomas Telford Publishing, 1997, https://doi.org/10.1680/cc.25929.

[82] A.M. Neville, Properties of concrete-5th edition, 2011.

[83] M. Liang, S. He, Y. Gan, H. Zhang, Z. Chang, E. Schlangen, B. Šavija, Predicting micromechanical properties of cement paste from backscattered electron (BSE) images by computer vision, Mater. Des. 229 (2023) 111905, https://doi.org/10.1016/j.matdes.2023.111905.

[84] B. Šavija, H. Zhang, E. Schlangen, Micromechanical testing and modelling of blast furnace slag cement pastes, Constr. Build. Mater. 239 (2020), https://doi.org/10.1016/j.conbuildmat.2019.117841.

[85] K. He, X. Zhang, S. Ren, J. Sun, Deep Residual Learning for Image Recognition, (2015).

[86] Y. Wu, K. He, Group Normalization, (2018).

[87] S. Elfwing, E. Uchibe, K. Doya, Sigmoid-Weighted Linear Units for Neural Network Function Approximation in Reinforcement Learning, (2017).

[88] K. Simonyan, A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, (2014).