# Microstructure-informed deep convolutional neural network for predicting short-term creep modulus of cement paste

Liang, Minfei; Gan, Yidong; Chang, Ze; Wan, Zhi; Schlangen, Erik; Šavija, Branko

**DOI**
10.1016/j.cemconres.2021.106681

**Publication date**
2022

**Document Version**
Final published version

**Published in**
Cement and Concrete Research

**Citation (APA)**
Liang, M., Gan, Y., Chang, Z., Wan, Z., Schlangen, E., & Šavija, B. (2022). Microstructure-informed deep convolutional neural network for predicting short-term creep modulus of cement paste. *Cement and Concrete Research*, *152*, 1-17. Article 106681. https://doi.org/10.1016/j.cemconres.2021.106681

Cement and Concrete Research 152 (2022) 106681

Contents lists available at ScienceDirect

Cement and Concrete Research

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

# Microstructure-informed deep convolutional neural network for predicting short-term creep modulus of cement paste

Liang Minfei, Gan Yidong *, Chang Ze, Wan Zhi, Schlangen Erik, Šavija Branko

*Microlab, Faculty of Civil Engineering and Geosciences, Delft University of Technology, Delft 2628, CN, the Netherlands*

ARTICLE INFO

*Keywords:*
Cement paste
Creep
Prediction
Convolutional neural network
Image processing

ABSTRACT

This study aims to provide an efficient alternative for predicting creep modulus of cement paste based on Deep Convolutional Neural Network (DCNN). First, a microscale lattice model for short-term creep is adopted to build a database that contains 18,920 samples. Then, 3 DCNNs with different consecutive convolutional layers are built to learn from the database. Finally, the performance of DCNNs is tested on unseen testing samples. The results show that the DCNNs can achieve high accuracy in the testing set, with the $R^2$ all higher than 0.96. The distribution of creep modulus predicted by the DCNNs coincides with that of the original data. Furthermore, through analyzing the feature maps, it is found that the DCNNs can correctly capture the local importance of different microstructural phases. The DCNN allows therefore prediction of the creep modulus based on microstructural input, which saves computational resources of segmentation procedure and multiple incremental FEM calculations.

## 1. Introduction

Creep of cement paste is one of the main reasons accounting for time-dependent deformation of concrete structures. For concrete structures at early age, creep can prevent early age cracking because it largely reduces shrinkage-induced tensile stress [1,2]. In the long run, creep is undesirable because it may cause excessive deflection and loss of prestress [3,4]. Therefore, evaluation of serviceability and durability of concrete structures requires accurate prediction of their creep behavior.

In recent years, a number of experimental studies have been conducted to investigate influences of multiple factors on creep of cementitious materials, including different mixture proportions [5–8], hydration degree [9–11], environmental conditions [12,13] and loading schemes [14,15]. Moreover, based on experimental results, numerical models were developed to simulate the creep or relaxation behavior with emphasis on different influencing factors (i.e., relative humidity [16], hydration degree [17], loading eccentricity [18]) or different numerical implementation of constitutive models (i.e., rate-type creep model [19–21], time-varying generalized Maxwell model [22], age adjusted effective modulus [23], and parallel creep curve [24].) Although these studies delivered in-depth understandings of creep or relaxation behavior, accurate prediction remains a complex issue because of gaps in length scales, oversimplification of material heterogeneity and interaction between multiple influencing factors.

As a highly heterogeneous material, concrete properties can be characterized in length scales ranging from macroscale, mesoscale, microscale to nanoscale. Studies at different scales adopt different assumptions: 1) at the macroscale, concrete is considered as homogenous in order to compute the response of the whole structure to external actions; 2) at the mesoscale, concrete is assumed to be a three-phase material constituted by cement paste, aggregate and interfacial transition zone (ITZ); 3) at the microscale, cement paste is considered an assembly of hydration products, unhydrated cement particles, and capillary pores; 4) at the nanoscale, the hydration products such as calcium silicate hydrate (C-S-H) with certain molecular structure are considered as "building blocks" of the material. It is clear that studies at finer scale adopt assumptions that are closer to intrinsic mechanisms of the material. However, the compromise between finer scale and higher scale should be made to achieve computational efficiency. Thereafter, multiscale studies are significant for bridging the finer scale models with the higher scale ones, in which finer scale models provide parameters for higher scale models. In the context of creep modelling, studies [25–27] prove the significant effects of cement paste on creep of concrete, while aggregate shows only elastic response. Therefore, an accurate

* Corresponding author.
*E-mail addresses:* M.Liang-1@tudelft.nl (L. Minfei), Y.Gan@tudelft.nl (G. Yidong), Z.Chang-1@tudelft.nl (C. Ze), Z.Wan-1@tudelft.nl (W. Zhi), Erik.Schlangen@tudelft.nl (S. Erik), B.Savija@tudelft.nl (Š. Branko).

https://doi.org/10.1016/j.cemconres.2021.106681
Received 11 August 2021; Received in revised form 11 November 2021; Accepted 24 November 2021
Available online 3 December 2021

characterization of creep behavior of cement paste is an essential part for comprehensive multi-scale modelling.

Thanks to the nano- and micro- indentation techniques, amounts of experimental and numerical studies on viscous property of cement paste have been made possible, which measured the intrinsic creep properties of two kinds of C-S-H and porous structures [28,29]. Recently, Gan et al. [30–33] proposed the methodology of micro-cantilever bending test to characterize the viscoelastic properties of cement paste, by which the effects of w/b ratio, binder type and stress level on creep were investigated. Their results showed that microstructural features largely influence the global creep behavior. Furthermore, based on experimental results of X-ray computed microtomography analysis and nano-indentation tests, Gan et al. [34] used a local- force based lattice fracture model to simulate the time-dependent deformation process of cement paste under constant load. The model was able to reproduce the experimentally observed viscous behavior of cement paste and reasonably explain the influence of microstructure on viscous properties. Besides, due to the abundant creep data gathered from microscale tests, many other microscale creep models were developed to investigate the influence of aggregate [35,36], C-S-H gel [37] and hydration [38]. Despite these models providing in-depth explanations for viscous mechanisms of cement paste and offering an easier way to predict viscous properties of cement paste, the complexities brought by the highly heterogenous microstructure still put burdens on computational capabilities, which can cause inefficiency especially when large amounts of computations need to be carried out.

As a standout image identification approach of Machine Learning (ML), Deep Convolutional Neural Network (DCNN) has been broadly used in the context of structural health monitoring. DCNNs have been shown to have superior performance for detection and segmentation of concrete cracks [39–45], which enabled development of intelligent applications for automatic crack detection. DCNNs have also been used for automatic inspection and evaluation for structural health [46–49]. However, only a few researches have been concentrated on the potential of DCNN in predicting properties of heterogeneous materials. Rao et al. [50] proposed a DCNN to predict the effective stuffiness and Poisson's ratio of representative volume elements (RVEs) with random inclusions, which showed significantly higher computational efficiency than conventional numerical homogenization approaches. Similarly, Yang et al. [51] implemented DCNNs and principal component analysis (PCA) to predict the stress-strain curve of a randomized composite microstructure. They derived a mean absolute error (MAE) of less than 10% of the range of values in a small training set, which showed the potential of DCNN in efficient material design. However, there is no study concentrating on how DCNNs improve computational efficiency for property homogenization of cement paste.

In light of the computational inefficiency of traditional FEM models, this study explores the feasibility of DCNN as an alternative for predicting the viscous property of heterogenous cement paste using microstructural images as input. Firstly, based on a previously developed microscale lattice fracture model, creep strain curves of 18,920 microstructures derived from X-ray computed tomography (XCT) will be generated. Then, well-constructed DCNNs will be trained based on the database. Finally, performance of the DCNN will be evaluated by multiple metrics and parametric studies on unseen data will be implemented.

## 2. Microscale lattice creep model

As a data-driven approach, the performance of DCNN strongly depends on the size and reliability of the source database. This study implements an experimentally-validated micro-scale lattice model to generate the database of creep behavior of cement paste with different microstructures [34]. The workflow of the lattice model and DCNN is shown in Fig. 1. Firstly,18,920 XCT images of the size 200 by 200 μm are segmented to generate corresponding random lattice networks, which

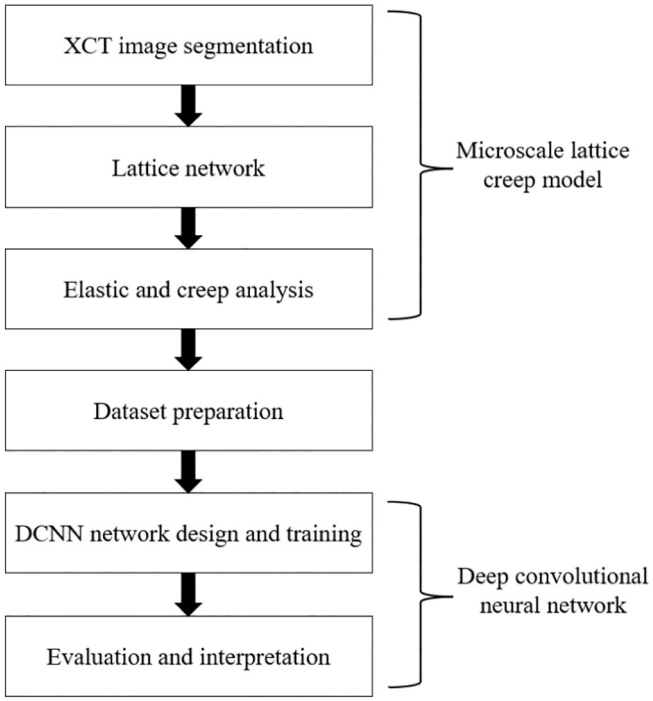



**Fig. 1.** Workflow of lattice creep model and DCNN.

are further discretized into 4 phases: High-density CSH (HD-CSH), low-density CSH (LD-CSH), unhydrated particles and pores. Then, after a cycle of elastic analysis, the method of local force is implemented to simulate the creep effects of HD-CSH and LD-CSH, respectively. After the predefined creep cycles are achieved, the creep compliance curve is fitted according to power function law and creep modulus of different microstructures can be derived, which characterizes the creep behavior of the cement paste and forms the training database for the DCNNs.

### 2.1. Generation of virtual specimens

To build a database that describes creep behavior of cement paste with different microstructure, 9460 slices of 200 μm *200 μm 2D microstructure are cropped from XCT images of 3 micro-cantilever beams of the size 1650um* 300um*300um by 3 directions: left-right, front-back and top-down. Details on the fabrication process of the micro-cantilever beams can be found in the authors' previous work [30]. Afterwards, each image is segmented into four phases (i.e., LD-CSH, HD-CSH, pores, and unhydrated particles) according to the grey scale histogram of the 9460 XCT images, as shown in Fig. 2. The segmentation of HD CSH, LD CSH, pore and unhydrated cement is done by the 3 threshold grey scale values $S_1$, $S_2$, $S_3$, which are calculated according to the global thresholding method [52]. The inflection of the CDF was defined as the upper threshold value for pores ($S_1$). The grey scale value corresponding to a change of the tangent slope of PDF was used as the lower threshold level for UHC ($S_2$). The division between the LD-CSH and HD-CSH is determined according to a relationship between hydration degree and w/c ratios [53], which is expressed as:

$$M_r = 3.017\left(\frac{w}{c}\right)\alpha - 1.347\alpha + 0.538 \tag{1}$$

where $M_r$ is the ratio of the mass of LD-CSH to HD-CSH; α is the hydration degree, which can be estimated by the following equation:

$$\alpha = \frac{\frac{V_h}{V}}{\frac{V_h}{V} + V_u} \tag{2}$$

where $V_h$, $V_u$ are volume fraction of hydration product and unhydrated

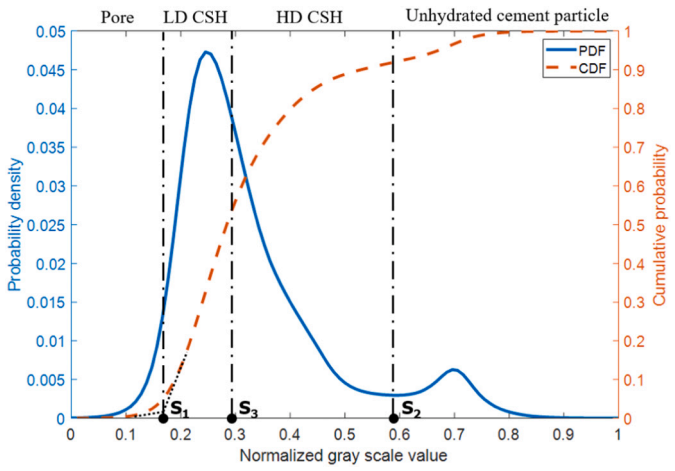



**Fig. 2.** Segmentation by grey scale histogram of XCT images.

particle respectively; V is the volume ratio of reaction product to reactant, which can be assumed as 2.2 according to [54]. Thereby, the threshold values $S_1$, $S_2$, $S_3$ for distinguishing the four phases can be calculated, as shown in Fig. 2.

To introduce local heterogeneity and disorder to the model, an irregular lattice network is generated as shown in Fig. 3. Firstly a 200*200 array of square cells with the side length 1 um is generated and each cell is assigned with different element type according to the segmentation results. Then sub cells with a side length ratio of 0.5 [55] are generated and positioned in the center of each cell. Within each sub cell, the nodes of the lattice beam elements are positioned randomly using a pseudo-random number generator following a uniform distribution. Afterwards, based on the random beam nodes array, Delaunay triangulation is performed to connect the neighboring nodes and form the lattice network [56].

Following the steps introduced above, 9460 4-phase lattice virtual specimens corresponding to different microstructures are generated.

### 2.2. Simulation of short-term creep behaviors

4 randomly-selected XCT scans, corresponding virtual specimens and boundary conditions are shown in the Appendix A1. The short-term creep behavior of cement paste is simulated under the boundary condition of axial compression test. It should be noted that in this study, the 2D simulations are conducted as a simplification, which cannot consider the realistic 3D connectivity and tortuosity of pore structure. Although the 2D simulations tends to produce lower elastic modulus in elastic analysis [57], similar average prepeak behaviors as in 3D simulations can still be characterized in 2D [58]. For creep simulation, if the connectivity of viscoelastic phase (CSH matrix) is preserved and the volume

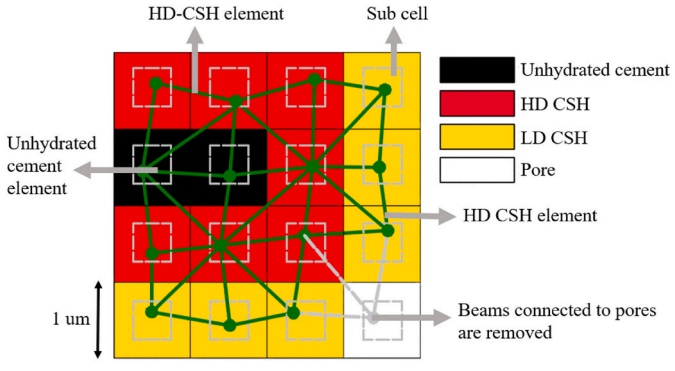



**Fig. 3.** Irregular lattice network.

of the elastic inclusions is relatively low, the validity of 2D model as simplification of 3D can still hold [59,60]. In view of the numerical convenience and computational efficiency, 2D model is adopted in this study to efficiently generate the dataset for the training of the DCNN.

Local properties of different phases, including the tensile strength $f_i$, elastic modulus $E$ and creep modulus $C$ are derived from the authors' previous testing results [30,32,33], as summarized in Table 1. Note that according to conventional nanoindentation test results, the mechanical properties of the four phases at a scale smaller than 5 μm are not dependent on w/c ratios [61,62].

With the boundary conditions and local material properties settled, an elastic calculation is firstly performed to get the initial mechanical response of the microstructure to the constant load. During this process, the beam elements that meet maximum stress criterion will be removed, which characterizes the damage induced by the immediate constant load. According to [63,64], the fracture stress can be calculated as follows:

$$\sigma_f = \alpha_N \frac{N}{A} + \alpha_M \frac{M}{W} \tag{3}$$

Assuming the cross-section of the beam element is circular, the notations in Eq. (3) are as follows: $A$ is the cross-sectional area of every beam element; $W$ is the cross-sectional moment of resistance of every beam element ($W = \pi D^3/32$, where D is the effective diameter of the lattice element.); $N$ and $M$ are the axial force and bending moment of the beam element; $\alpha_N$ and $\alpha_M$ are the influencing factors of axial force and bending moment, which normally equal 1.0 and 0.05, respectively, according to [65–67].

After the initial mechanical response of the microstructure to the immediate constant load is calculated, the creep calculations are conducted using an incremental local force method [68]. In each following step, a local force is calculated based on the local mechanical response of each node (calculated in first step) to mimic the short-term creep effect. First, the creep compliance function of CSH is expressed as:

$$C(t, t_0) = \frac{1}{E_c}\left(\frac{t - t_0}{\Delta t}\right)^{\beta} \tag{4}$$

In which $C(t, t_0)$ is the creep compliance at time t when loaded at time $t_0$; $\Delta t$ is the time interval used in the calculation, which is 1 s in this study; $E_c$ is the creep modulus of the CSH phases (Table 1); $\beta$ is a constant which equals to 0.251 according to [69]. Then, based on the Euler's method, the incremental creep strain at each time step can be calculated as:

$$\varepsilon_{i+1}^{cr} - \varepsilon_i^{cr} = \frac{1}{E_c}\beta \Delta t \sigma_i \left(\frac{t - t_0}{\Delta t}\right)^{\beta - 1} \tag{5}$$

in which $\varepsilon_i^{cr}$ is the creep strain at $i$-th step; $\sigma_i$ is the local stress. Thereby, the local creep strain of each beam is dependent on the local stress distribution, which is converted to local force [68]. These local forces applied on CSH beam elements result in overall creep of the whole microstructure. The local stress distribution of the 4 randomly-selected virtual specimens at time step $t = 30$s is shown in Appendix A2. Based on the methods introduced above, the short-term creep behavior of a specific microstructure of cement paste can be calculated. For the 4 virtual specimens shown in Appendix A1, their results of creep compliance under 30 s' loading are shown in Appendix A3. By fitting the creep compliance curves with Eq. (4), the creep modulus can be derived. The fitted exponent β equals to 0.4285 and remains constant for all

**Table 1**
Local properties of different phases.

| Phases | $f_i$/ MPa | E/ GPa | C/ GPa |
|---|---|---|---|
| Unhydrated Cement | 614.7 | 84.2 | – |
| LD CSH | 52.2 | 21.3 | 670 |
| HD CSH | 82.8 | 26.4 | 990 |

specimens. According to the authors' experimental observations [30] and the modelling results of all other virtual specimens, the patterns of constant exponent and high coefficient of determination (higher than 0.995) remains valid. Thereby, only one parameter (i.e., creep modulus) is needed to represent the creep behavior of cement paste of certain microstructure.

## 3. Deep convolution neural network (DCNN)

Based on the microscale lattice creep model introduced in Section 2, the database for training the DCNNs can be produced. In this section, data derived from 18,920 virtual specimens will be used to train a self-developed DCNN, which will form an efficient approach for predicting creep behavior of cement paste with different microstructure.

### 3.1. Database

#### 3.1.1. Data augmentation

As a data-driven approach, the performance of the DCNNs strongly depends on the amount of data. In this study, a data augmentation procedure is performed by rotating the input microstructure. As is shown in Fig. A1, the presumed boundary condition of the model is axial compression, so the orientation of the specimen will strongly influence the modelling result because of the heterogeneity of microstructure. For example, the creep modulus of the 4 specimens in Fig. A3 will be changed from 5614, 5796, 9361, 7059 GPa to 6870, 7570, 6350, 6170 GPa, if they are rotated 90 degrees anticlockwise. Therefore, by rotating the input microstructure and running the lattice creep model with the same boundary condition, different creep behaviors will be derived. Note that due to the symmetry of boundary conditions and microstructure, the rotating angle of 0 degrees and 180 degrees will produce the same results, as well as the rotating angle of 90 degrees and 270 degrees. In this case, only rotating angle of 0 and 90 are chosen to produce the whole database for the training of DCNNs. With the 9460 XCT images of different microstructure, there are in total 18,920 sets of data in this database.

#### 3.1.2. Overview of the database

Based on the microscale lattice model, a database that contains 18,920 sets of data is established, with each set including an image of a specific microstructure and a corresponding creep modulus. The overview of the database is shown in Fig. 4. 5 parameters are selected here to describe the database: with one parameter (i.e., creep modulus $E_c$) being the modelling result that characterizes the creep behavior of cement paste and four other parameters (i.e., poro (porosity), LDR (LD/CSH ratio), DOH (Degree of hydration), UHC (Unhydrated cement particle ratio)) characterizing the constituent of the corresponding microstructure. Based on the segmentation of the input XCT image, the four parameters that related to the microstructure are calculated as follows:

$$poro = \frac{N_{pore}}{N_{total}} \tag{6-a}$$

$$LDR = \frac{N_{LD-CSH}}{N_{LD-CSH} + N_{HD-CSH}} \tag{6-b}$$

$$DOH = \frac{N_{LD-CSH} + N_{HD-CSH}}{N_{total}} \tag{6-c}$$

$$UHC = \frac{N_{uhc}}{N_{total}} \tag{6-d}$$

where $N_{total}$ is the number of image pixels, which equals to 40,000 in this study; $N_{pore}$, $N_{LD\text{-}CSH}$, $N_{HD\text{-}CSH}$, $N_{uhc}$ refer to the number of lattice cells that represent pores, LD-CSH, HD-CSH and unhydrated cement particles respectively. In Fig. 4(a), the diagonal figures show the histogram of each parameter and the other figures show the correlation between any pair of parameters. Fig. 4(b) shows the exact correlation between any two parameters. From Fig. 4, clear correlations between the creep modulus and the other four parameters can be found: the creep modulus decreases with porosity, LDR and DOH, while it increases with UHC. Besides, a strong correlation can also be found between DOH and UHC, while LDR and porosity show little correlation with DOH. Note that although the correlations of creep modulus with DOH and UHC are much weaker than that with LDR and porosity, it is contradicted to common understandings. According to common understandings, higher DOH and lower UHC means a more mature and integrated CSH microstructure, whose creep modulus should be higher. However, the Lattice model gives an opposite trend. This difference is because the stiffness of unhydrated cement particle is much higher than other phases (see Table 1), and the lattice model adopted here does not consider the aging effects of other material phases. In other words, the lattice model calculates the creep behavior of cement paste totally from the perspective of microstructural mechanics and therefore its results regarding hydration effects are not formal.

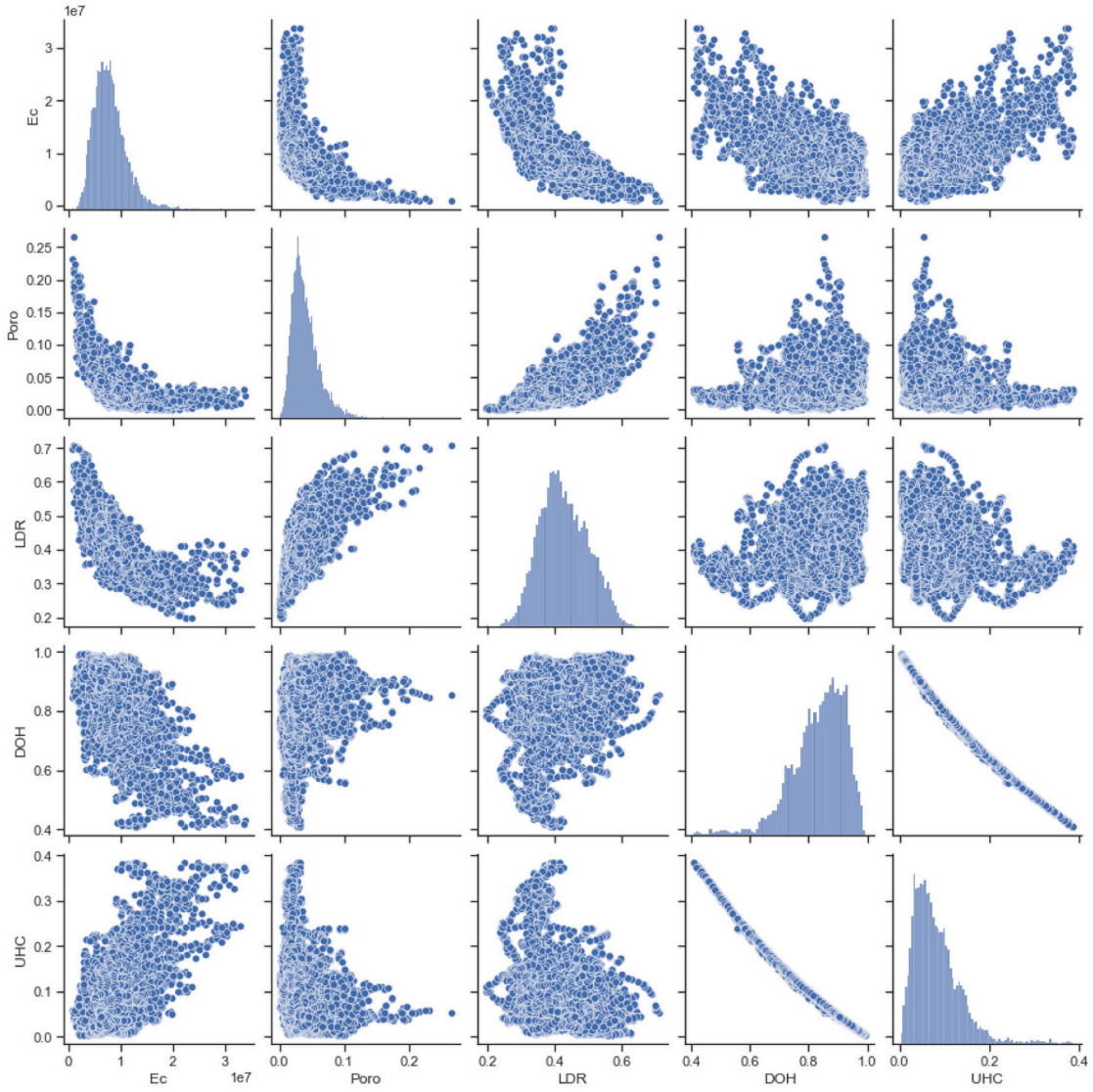
### 3.2. DCNN architecture

In this study, 3 kinds of DCNN architecture with 1 single convolutional layer, 2-stack consecutive convolutional layers, and 3-stack consecutive convolutional layers will be adopted, which are named as DCNN-1, DCNN-2 and DCNN-3, respectively. Taking DCNN-3 as an example, its architecture is shown in Fig. 5, and a detailed description of its configuration are listed in Table 2. The DCNN is formulated by the following 6 kinds of layers: convolutional layer, max pooling layer, ReLU activation layer, global pooling layer, fully-connected dense layer and linear activation layer. As is shown in Fig. 5, firstly, the input layer receives a grayscale XCT image representing a specific microstructure. Then, the image is passed through a series of consecutive convolutional layers and processed by a ReLU activation function and a max pooling layer. Subsequently, the stacked feature maps produced by the convolutional layers will be flattened by a global pooling layer and transferred into a 512*1 vector. The vector will then be passed through fully-connected dense layers and finally output a scalar, which is the creep modulus that can describe the creep behavior of the input microstructure under axial compression (see Fig. 4).

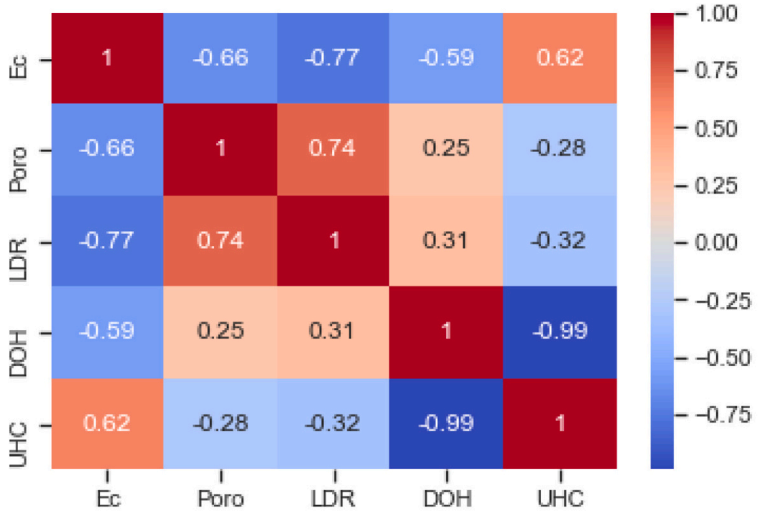
#### 3.2.1. Consecutive convolutional layer

The convolutional layer of DCNN is what make it stand out from other ML algorithms when processing tasks of image recognition. An illustration of the convolution operation used in this study is shown in Fig. 6. As shown by Fig. 6(a), the convolution operation is performed by sliding a filter (i.e., kernel) over the whole input image step by step. Within each sliding step, the part of the input image covered by the filter is called the receptive field. Note that the filter is a matrix of trainable parameters whose size is much smaller than that of the input image, while the receptive field is a matrix of the grey scale value, whose size is the same as that of the filter. As shown by Fig. 6(b), in every step, by calculating the dot product of the filter matrix and the receptive field matrix, the local feature of the input image is extracted to form a new input image for next layers. This new input image is then activated by a non-linear function and then form a feature map (i.e., activation map), which will be passed to the subsequent convolutional layers. Note that the illustration in Fig. 6 only considers an input grey-scale image with only one channel. For image with multiple channels, which happens before arriving at the second convolutional layers (see Fig. 5), the filter has the same depth as the channel of the input image. Besides, to ensure the consistency of image size, empty rows and columns are padded on the edge of the output image after each convolution operation.

As mentioned above, the filter plays a significant role in the process of convolution operation. A larger filter means a larger receptive field, which will extract more information at each step. However, simply increasing the filter size will bring in too many parameters and raise a




(a) Histogram and correlation scatters




(b) Correlation matrix

**Fig. 4.** Overview of the database.

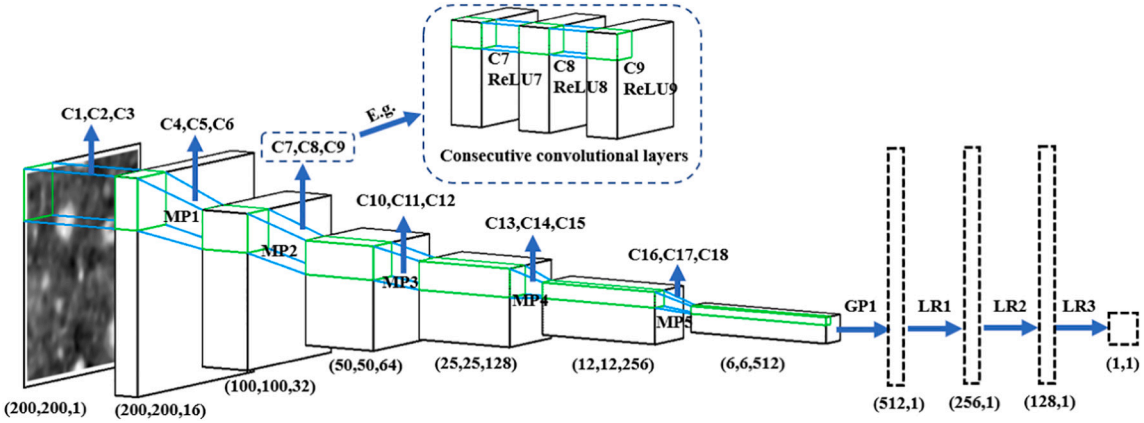



**Fig. 5.** DCNN-1 architecture
(C#: Convolutional layer; MP#: Max pooling layer; ReLU#: ReLU activation function; GP#: Global average activation function; LR#: Linear activation function).

**Table 2**
Detailed configurations of the DCNN-1.

| Consecutive Layer # | Layer # | Layer type | Output shape | Number of Parameters |
|---|---|---|---|---|
| | 1 | Input | 200*200*1 | – |
| | 2 | Conv1 | 200*200*16 | 160 |
| 1 | 3 | Conv2 | 200*200*16 | 2320 |
| | 4 | Conv3 | 200*200*16 | 2320 |
| | 5 | Max pooling1 | 100*100*16 | 0 |
| | 6 | Conv4 | 100*100*32 | 4640 |
| 2 | 7 | Conv5 | 100*100*32 | 9248 |
| | 8 | Conv6 | 100*100*32 | 9248 |
| | 9 | Max pooling2 | 50*50*32 | 0 |
| | 10 | Conv7 | 50*50*64 | 18,496 |
| 3 | 11 | Conv8 | 50*50*64 | 36,928 |
| | 12 | Conv9 | 50*50*64 | 36,928 |
| | 13 | Max pooling3 | 25*25*64 | 0 |
| | 14 | Conv10 | 25*25*128 | 73,856 |
| 4 | 15 | Conv11 | 25*25*128 | 147,584 |
| | 16 | Conv12 | 25*25*128 | 147,584 |
| | 17 | Max pooling4 | 12*12*128 | 0 |
| | 18 | Conv13 | 12*12*256 | 295,168 |
| 5 | 19 | Conv14 | 12*12*256 | 590,080 |
| | 20 | Conv15 | 12*12*256 | 590,080 |
| | 21 | Max pooling5 | 6*6*256 | 0 |
| | 22 | Conv16 | 6*6*512 | 1,180,160 |
| 6 | 23 | Conv17 | 6*6*512 | 2,359,808 |
| | 24 | Conv18 | 6*6*512 | 2,359,808 |
| | 25 | Global average pooling1 | 1*512 | 0 |
| | 26 | Dense1 | 1*256 | 131,328 |
| | 27 | Dense2 | 1*128 | 32,896 |
| | 28 | Dense3 | 1*1 | 129 |
| | Total parameters | | | 8,028,769 |
| | Trainable parameters | | | 8,028,769 |
| | Non-trainable parameters | | | 0 |

problem of computational inefficiency. Therefore, this study configures a 3-stack consecutive convolutional layer instead of a single one. Such net architecture can enable a larger receptive field while only using small filter size [70]. A similar application can also be found in other well-established DCNN architectures like Alex Net [71]. The explanation is shown in Fig. 6 (c). For a single convolutional layer, given the filter size of 3*3, one can only get 3*3 receptive fields like A, B, C in the left of Fig. 6 (c). However, when a second convolutional layer is applied, the information contained in B is incorporated into A, which includes information that lies outside of A in the first layer. As this happens in every direction, the receptive field of A is augmented from 3*3 to 5*5, as is shown in the right part of Fig. 6 (c). And if a third layer is applied, the receptive filed of C is also incorporated by A, which then enlarges A's receptive field to 7*7. Therefore, by stacking consecutive convolutional layers, the receptive field can be effectively augmented without increasing the filter size. In this study, a single convolutional layer with a filter size of 3*3 and a stride size of 1 is adopted.

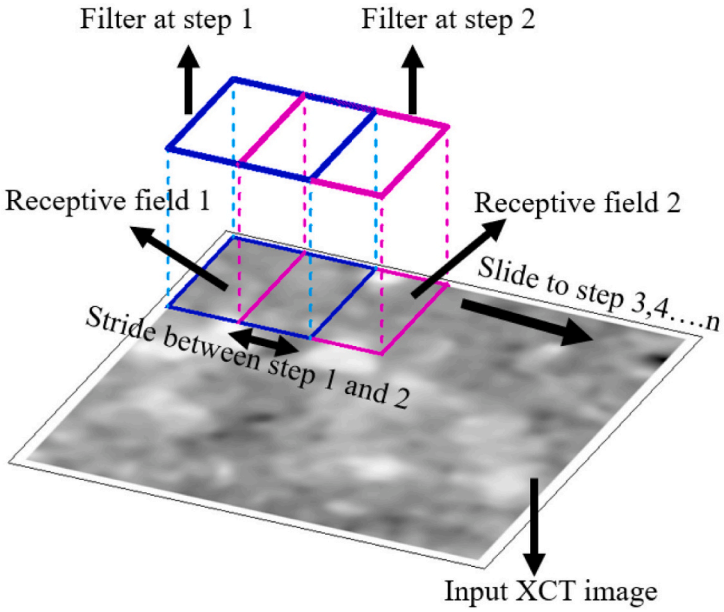
### 3.2.2. Pooling layer

Following the convolutional layers, pooling layer is another component that distinguishes CNN from conventional neural networks. After the image is processed by the convolutional layers, its dimension is largely augmented. For example, in this study, after the first consecutive convolution layers (see Fig. 5 or Table 2), the image size is augmented from 200*200*1 to 200*200*16. Such a large increase of image dimension puts a heavy burden on the computational capabilities. In light of this situation, the pooling layer is incorporated in the DCNN architecture to fulfill 2 objectives: 1) Reduce the dimension of image array; 2) Reserve the key information extracted by the convolutional layers. An illustration of how the pooling layer works is given in Fig. 7. Similar to the convolution operation, in the pooling process, a pooling filter will slide over the input feature map, which is delivered by the last convolutional layer. In every step, the area corresponding to the pooling filter is called the pooling field. Then the statistics (i.e., max or average) of all elements in the pooling field matrix will be calculated and form the downscaled feature map. Currently, there are mainly two kinds of pooling layers that are broadly adopted in various CNN applications: Max pooling and average pooling. Furthermore, as a more discriminative approach, max pooling has proven to be more effective than average pooling [72]. Therefore, in this study, max pooling layers with pooling size 2*2 and pooling stride 2 are adopted to be attached to each consecutive convolutional layer.

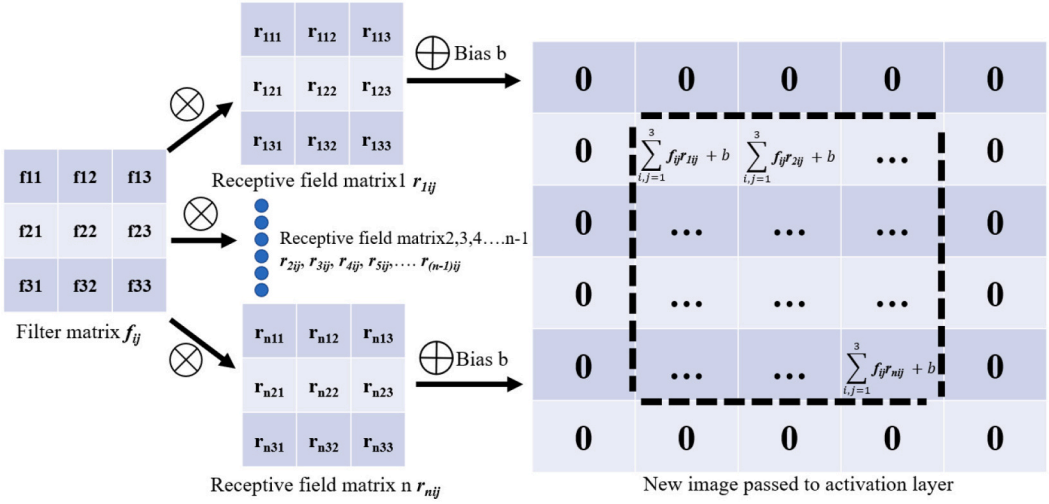
After all convolution operations are finished, a global average pooling layer [73] is adopted. The global average pooling layer takes the average of the feature map at each channel and forms a 1D vector to feed the next dense layer. For example, when the output image block (6*6*512) at the 24th layer is passed to the 25th layer, its dimension is reduced to 1*1*512, with the 512 elements representing the global average value of all 6*6 feature maps of the 6*6*512 image block.

### 3.2.3. Activation layer

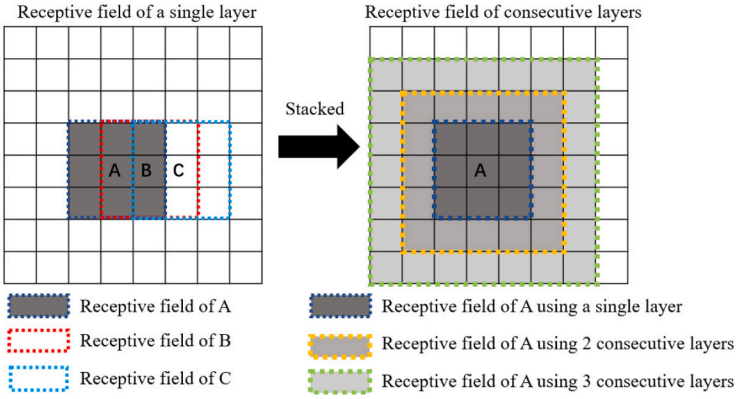
Activation layer is a key component in Artificial Neural Network (ANN) and CNN, which introduces nonlinearity to the linear transformation process. In typical ANN, activation functions like sigmoid function ($f(x) = (1 + e^{-x})^{-1}$) and tanh function ($f(x) = tanh(x)$) are used. However, such highly-nonlinear functions can cause computational inefficiency and are not favorable for DCNN which requires much higher computational capabilities than conventional ANN. To tackle such problem, the Rectified Linear Unit (ReLU) function ($f(x) = \max(x,0)$)




(a) Filter slides across the input image




(b) Calculation flow




(c) Augmentation of receptive field by consecutive convolution layers

**Fig. 6.** Convolution operation.

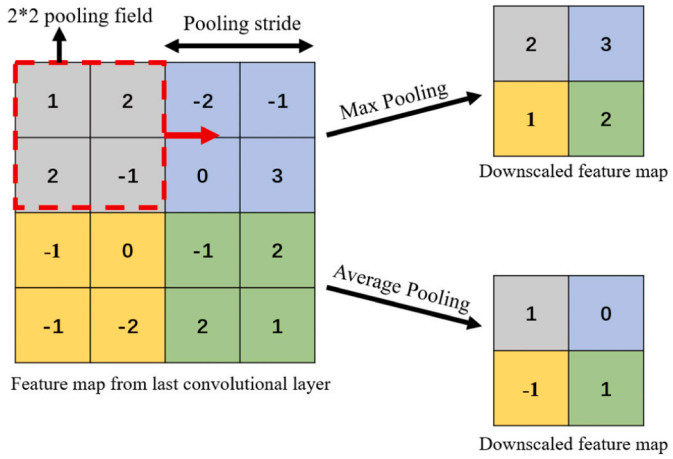



**Fig. 7.** Examples for max and average pooling layer.

[74] is introduced, which sets all negative input to zero and preserves all positive input. Unlike the sigmoid and tanh function, ReLU does not put limit on its output, and only produces gradients of 1 or 0, which effectively accelerates the training process. Therefore, in this study, the ReLU activation layer is adopted, which is attached after every single convolutional layer (see Fig. 5).

### 3.3. Training configuration

The training of DCNN can be conducted by importing the database (in Section 3.1) into the designed network architecture (in Section 3.2). Note that in this study, DCNN with a single convolutional layer (DCNN-1), 2-stack consecutive convolutional layers (DCNN-2) and 3-stack consecutive convolutional layers (DCNN-3) will be used to train the database in the meantime and their difference will be compared.

#### 3.3.1. Data preprocessing

Before training starts, the database will be shuffled and grouped into 3 sets: a training set, a validation set and a testing set following the proportion ratio 6.4: 1.6: 2. Thus, there are 12,108, 3028 and 3784 samples in the training, validation and testing set, respectively. Then, the data of creep modulus in training set will be standardized by its max and min value to make sure the range of each sample point lie in the range of (0,1). Note that the standardization should only be conducted within the training set and cannot be extended to the testing set, since the statistical information of the testing set should be totally isolated from training set.

#### 3.3.2. Optimization

During the training process, the weights and biases of the filters and dense networks are updated to optimize the prediction performance on training set. Mean squared error is adopted as the loss function. Then, by a gradient descent algorithm, the weights and filters are updated to minimize the loss function through backpropagation of error. In this study, the Adam algorithm (i.e., Adaptive Moment Estimation) [75] is adopted as the optimization function, which is an advanced version of RMSprop and Adadelta algorithms [76]. In the Adam algorithm, two momentum terms are incorporated to take the historical values of gradients into consideration, which can prevent the local minima and achieve high efficiency. The two momentum terms of Adam at t-th iteration are as follows:

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t \tag{7-a}$$

$$v_t = \beta_2 v_{t-1} + (1-\beta_2) {g_t}^2 \tag{7-b}$$

where $m_t$, $v_t$ are the first and second momentum terms; $g_t$ is the derivative of loss function; $\beta_1$ and $\beta_2$ are the decaying weight of the first and second momentum term, respectively. Because the initial values of the momentum terms are assumed as 0 in the first iteration, their values are normalized to get a bias-corrected momentum terms as follow:

$$\widehat{m}_t = \frac{m_t}{1-\beta_1^t} \tag{8-a}$$

$$\widehat{v}_t = \frac{v_t}{1-\beta_2^t} \tag{8-b}$$

Then the weights and bias of the DCNN at (t + 1)-th iteration can be updated by:

$$\theta_{t+1} = \theta_t - \frac{\alpha}{\sqrt{\widehat{v}_t} + \varepsilon} \widehat{m}_t \tag{9}$$

where $\theta_t$, $\theta_{t+1}$ denotes the targeted parameter at t and t-th iteration; $\alpha$ is the learning rate; $\varepsilon$ is the parameter for numerical stability, which equals to $1*10^{-8}$. In this study, the value of decaying weights $\beta_1$ and $\beta_2$ are set as 0.9 and 0.999, according to [71]. However, although given such an efficient optimization algorithm, the setting of learning rate is still based on presumptions. As learning rate determines the updating step of each iteration in the gradient descent process, an appropriate value of the learning rate is essential for the objective of finding the minima efficiently. Thereby, this study adopts an adaptive exponential-decay approach to update the learning rate: for every five epochs, if the coefficient of determination evaluated in validation set is not improved, the learning rate will be decayed by a proportion of 0.9. Note that during the training process, in every iteration, 32 samples are imported to the DCNN to update the variables. A complete training for all the images in training set is called an epoch. In this study, 200 epochs are conducted for the training of the DCNNs.

## 4. Results and discussion

### 4.1. Metrics and performance

#### 4.1.1. Training and validation

In this study, 3 metrics are used to evaluate the performance of the DCNNs in training, validation and testing set, which are Mean Squared Error (MSE), Mean Absolute Error (MAE) and Coefficient of Determination ($R^2$). They can be expressed as follows:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(x_i - \widehat{x})^2 \tag{10-a}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|x_i - \widehat{x}| \tag{10-b}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(x_i - \widehat{x})^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{10-c}$$

In the training process, MSE is adopted as the loss function that is used in gradient descent, and $R^2$ is used as the metrics for adjusting learning rate in every 5 epochs (as described in 3.3.2). Therefore, the MSE and $R^2$ are used to describe the training history. In the training and validation set, the MSE and $R^2$ of the 3 DCNNs are shown in Fig. 8 (a) ~ (c), and the learning rate history are shown in Fig. 8 (d). From Fig. 8, one can see that the 3 DCNNs all achieve high prediction performance in the training process. On the training and validation set, the $R^2$ of all the DCNNs are above 0.99 and 0.95, respectively, which indicates low risk of overfitting since the prediction performance of training and validation set are quite close. Besides, the results also prove the adaptive adjustment of learning rate are effective for the training process of DCNNs: for DCNN-1, DCNN-2 and DCNN-3, the learning rate decreases by 96.5%, 97.6%, and 97.8%

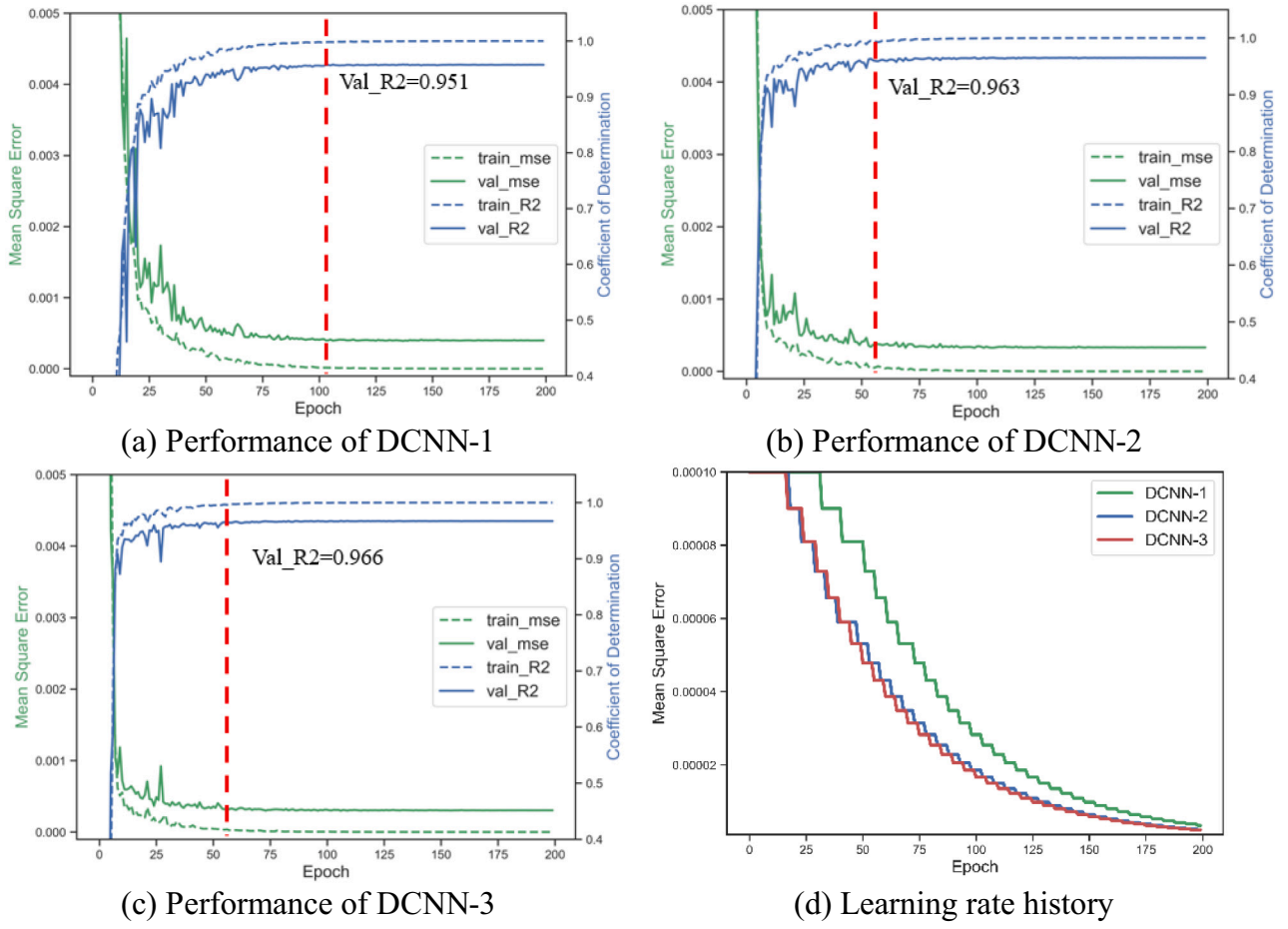



**Fig. 8.** Training history.

in 200 epochs, respectively. However, comparing the difference of DCNNs, one can find the importance of having consecutive convolutional layers. With the 3-stack convolutional layers, DCNN-3 gives the best prediction while maintaining the best numerical stability since the first iterations. Besides, the fast decrease of learning rate of DCNN-3 also indicates that better numerical stability is achieved. Comparing DCNN-1 with the others, more distinguished effects of consecutive convolutional layers on both prediction performance and numerical stability can also be found.

#### 4.1.2. Testing result

After the training process is finished, the DCNNs are tested by the 3784 data samples, which have been kept isolated since the beginning. The testing results of the DCNNs are shown in Fig. 9. The testing results of different DCNNs are consistent with that of the training and validation set: All the 3 DCNNs can achieve high accuracy over most samples within the unseen testing set. And more consecutive convolutional layers can improve the testing accuracy expressed by $R^2$, MSE and MAE.

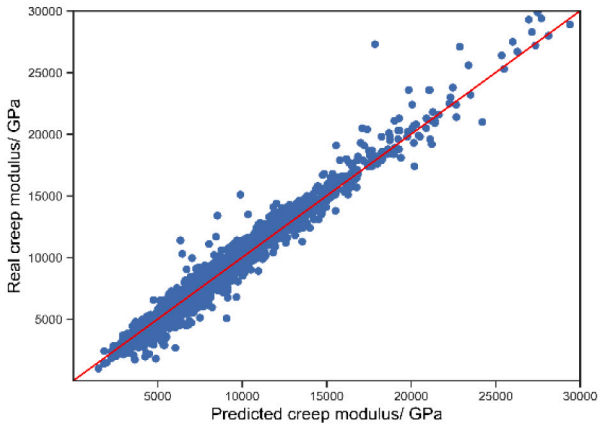
#### 4.1.3. Correlation with microstructural parameters

To further validate the capabilities of the DCNN, the correlation between the microstructural parameters and prediction results in the testing set are compared. According to Fig. 4, the most significant microstructure parameters that influence the creep modulus are LD-CSH ratio, Porosity and Unhydrated Cement Ratio. Therefore, the correlation between these parameters and creep modulus predicted by the DCNNs are shown in Fig. 10. By comparison, the distribution of creep modulus predicted by the DCNNs highly coincides with the original distribution of Lattice modelling results in testing set. Such distributions are also similar to the ones that shown in Fig. 4. These results further proves that the proposed DCNNs can effectively replicate the predictions made by the microstructural lattice model.

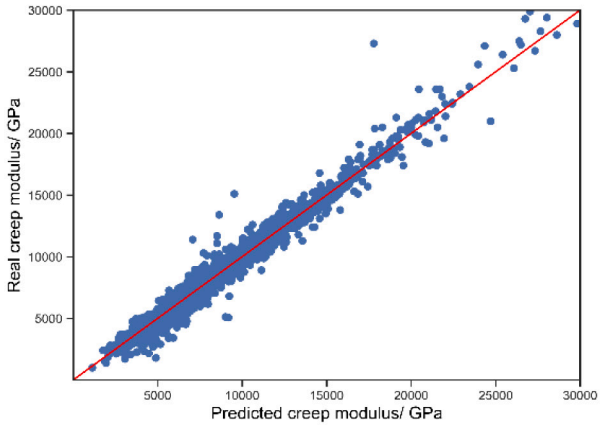
### 4.2. Feature map

This section discusses how the DCNNs capture the feature of an input microstructure and give its prediction. As mentioned in Section 3.2.1, during the convolution operation, the local feature of the microstructure is extracted by the filters to form a new input image block (a 3D matrix), which are called feature maps and are passed to the subsequent layers after ReLU activation and max pooling. Therefore, the feature maps indicate the local features identified by different convolutional layers. Taking DCNN-3 as an example, the feature maps produced at the end of every consecutive convolutional layer are shown in Fig. 11. Note that for each consecutive convolution layer, a stack of feature maps is produced. The results shown in Fig. 11 are the summation of these feature maps along its channel. Given the input microstructure (see Fig. 12(a)), the DCNNs keeps extracting the features and downscaling the size of feature maps. At the first layers, only the finer local features are extracted and compressed into smaller feature maps. While when it goes deeper into the DCNN, more details are compressed into smaller number of pixels and therefore such feature maps become highly abstract and uninterpretable. In further analysis for local importance of microstructure, the feature maps at the 4th layer will be used, which is close to the output layers and meanwhile maintains interpretability.

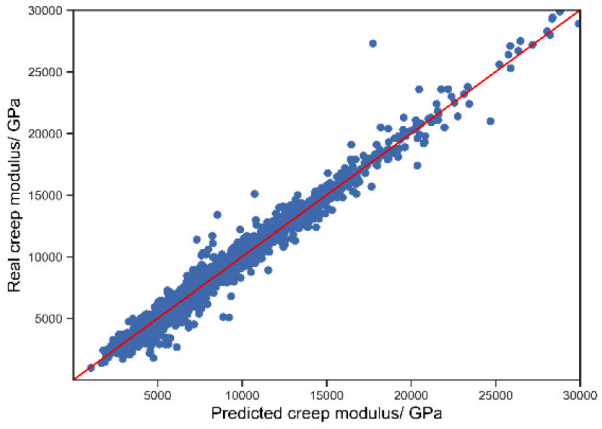
Except for sample 1, which is in consistent with Fig. 11, 3 other samples were randomly selected and imported to DCNN-3. Their feature maps at the 4th consecutive layer are summed along the channel and




(a) DCNN-1, $R^2$ = 0.9678, MSE = 0.00391, MAE = 0.01285




(b) DCNN-2, $R^2$ = 0.9753, MSE = 0.00300, MAE = 0.01100




(c) DCNN-3, $R^2$ = 0.9757, MSE = 0.00295, MAE = 0.01088

**Fig. 9.** Testing results.

shown in Fig. 12. By comparing the feature maps (right) with the input microstructure (left) and the segmentation result (middle), we can see that the phases of unhydrated cement particles and the surrounded HD-CSH are highlighted by the feature maps. Such distribution in the feature maps indicates that the phases of unhydrated cement particles and HD-CSH can cause higher creep modulus, which is consistent with the correlations that shown in Figs. 4 and 10. Besides, the feature maps tell that the phase of unhydrated cement particles is the most significant phase which increases the final prediction value of creep modulus. Such prediction reflects the inherence of the microscale lattice model. As can be seen from Table 1, the elastic modulus of unhydrated cement particles (614 GPa) is much higher than that of LD-CSH (52 GPa) and HD-CSH (82 GPa), so the incorporation of unhydrated cement particles actually increases the overall stiffness of the microstructure. Thereby, under the action of local force, smaller deformation at each creep step is obtained, which then gives a higher creep modulus in the fitting result. As can be seen from Figs. 4 and 10, there is a negative correlation between LD-CSH/ porosity and creep modulus, which is not explicitly shown here. This is because the ReLU activation function is used in every convolution operation, and the negative values produced during convolution operation are all replaced with zeros and only positive values are preserved. However, one can also see a complementary relationship between LD-CSH and HD-CSH, according to Eq. (6-b). By this relation, since HD-CSH is detected as a positive contribution to the magnitude of creep modulus, then the LD-CSH should be a negative one.

### 4.3. Further discussions

Sections 4.1 and 4.2 have shown that the DCNNs can accurately mimic the microscale lattice creep model. By comparison, the DCNNs save the computational resources for image segmentation and multiple incremental iterations of the microscale lattice creep model, and directly predict the creep modulus with the input of a microstructure represented by a raw XCT image. The capabilities of extracting and integrating local features from highly heterogeneous microstructures make DCNN a powerful tool in predicting the short-term creep of cement paste. Moreover, such capabilities indicate strong potential of DCNN in other computationally- intensive tasks for homogenization of heterogenous materials.

However, it should also be noted that the performance of the DCNNs highly depends on the microscale lattice creep model, which generated the datasets for the whole training process. The applications of DCNNs of this study is limited to the scenario of hardened four-phase cement paste under immediate axial compression. If any other phases (e.g., CH) of cement paste play an essential role, or other boundary conditions are to be considered, the microscale lattice creep model will need to be adjusted first. After new datasets that reflect the influence of new phases or different boundaries are generated, the DCNNs can be retrained to gain accuracy on the new tasks. If the changes of the microscale lattice creep model are minor, then transfer learning can be applied in this process. In this case, only parts of the DCNN network have to be adjusted to gain good accuracy for new datasets, which not only saves computational resources in training process but also tends to gain higher accuracy [77]. However, it should be noted that meaningful Transfer Learning requires that the new task should be related to the old task. Therefore, if significant changes are made to the inherent mechanisms of the microscale lattice creep model, such as the constitutive relationship, then the ML model will need to be retrained all over again.

## 5. Conclusions

In this study, we built a database that contains 18,920 microstructures and their corresponding creep moduli using an experimentally-validated microscale lattice model for short-term creep. Then, DCNNs with different numbers of consecutive convolutional layers are established to predict the creep modulus given the microstructure as input. Finally, the distribution of creep modulus predicted by DCNNs and the lattice model are compared, and the local importance of microstructure are analyzed based on the output of feature maps. Through this study, the following conclusions can be obtained:

(1) The modelling results of 18,920 microstructures show that the adopted microscale lattice model captures well the correlations between the creep behaviors and microstructural parameters,

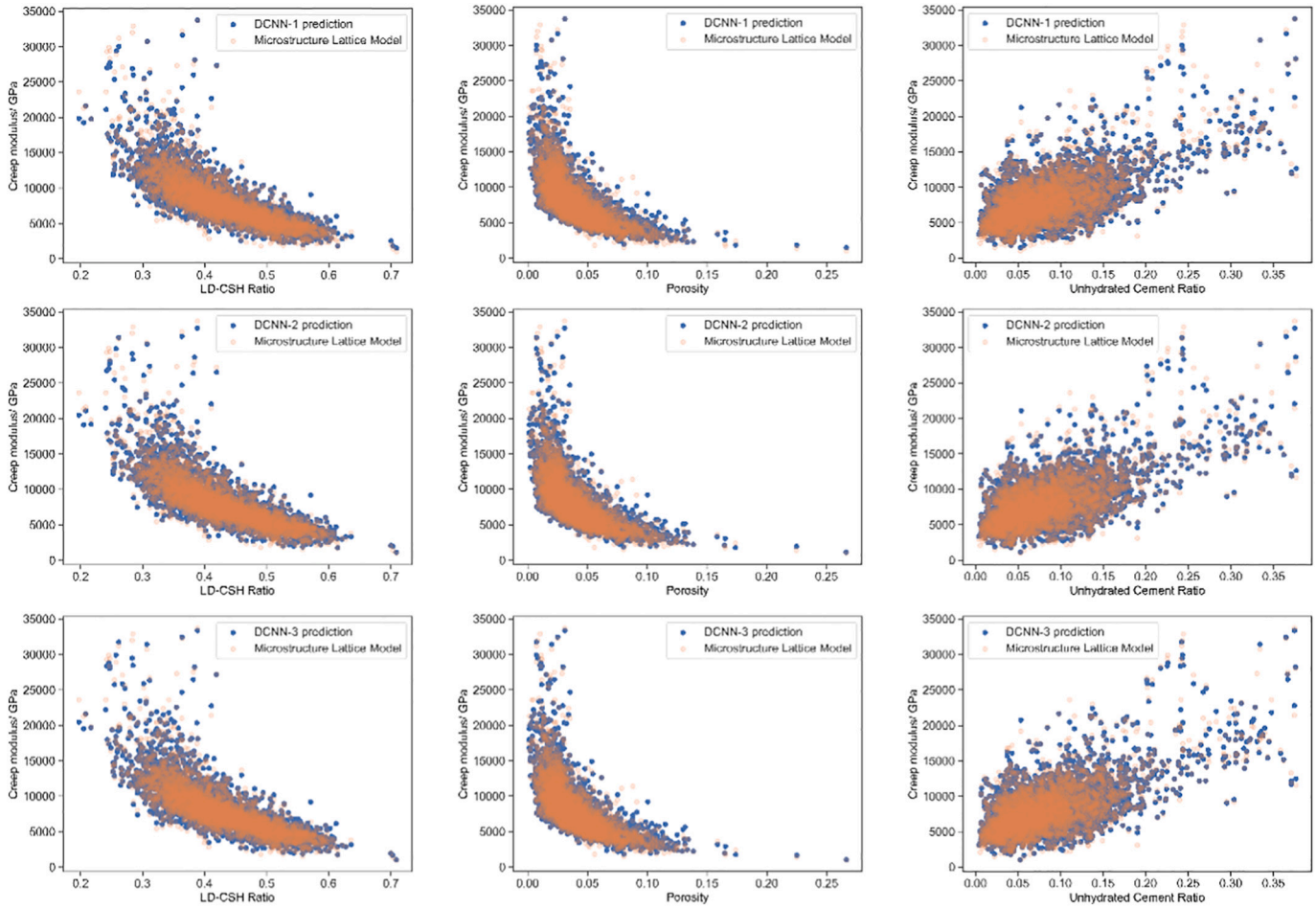



**Fig. 10.** Correlation between predicted creep modulus and microstructural parameters.

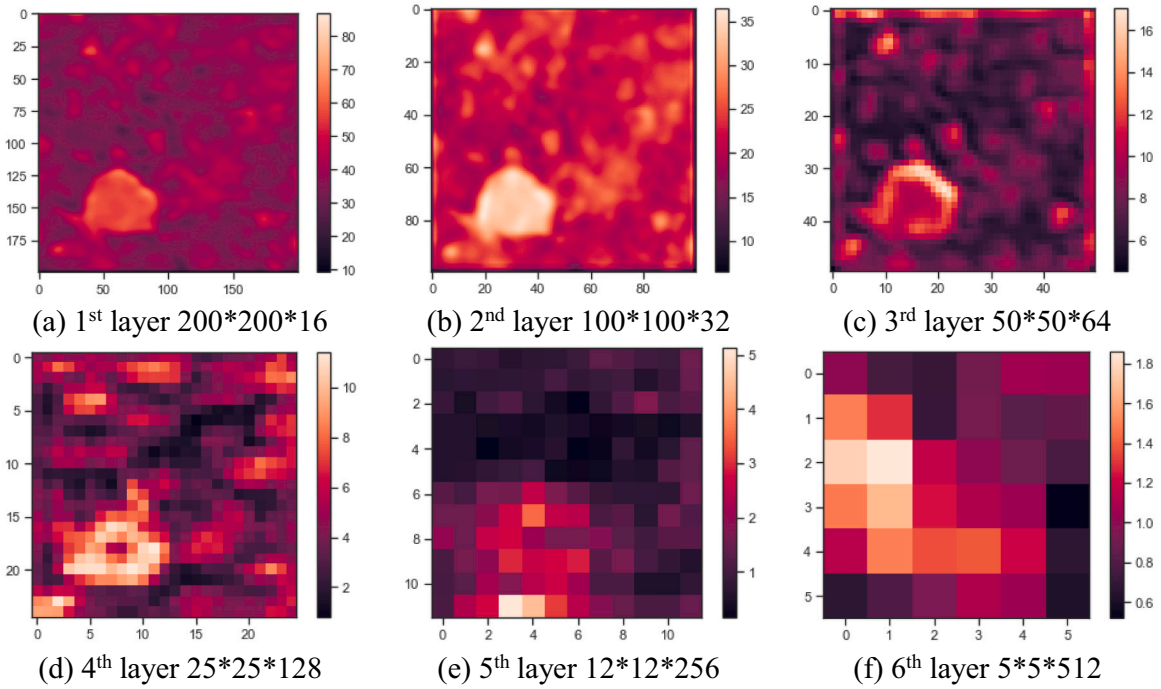



**Fig. 11.** Feature maps extracted from DCNN-3.

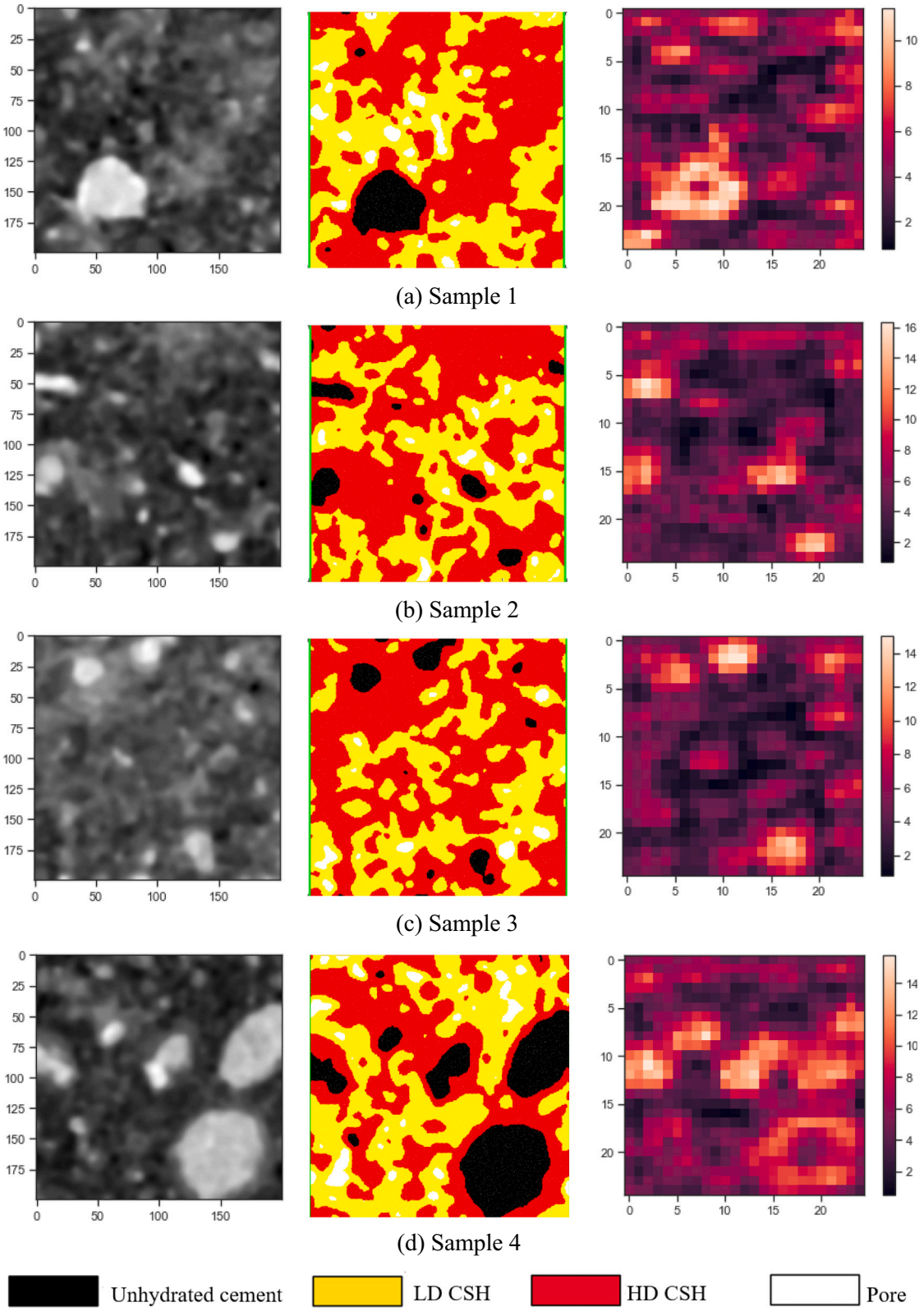



**Fig. 12.** Random selected samples for illustrating the feature importance
(Note: the images from left to right are: XCT image, Lattice modelling (segmentation result) and 4th layer feature map.)

including CSH content, porosity and unhydrated cement particle ratio.

(2) Due to the amplification of receptive field, the consecutive convolution layers can promote the prediction accuracy of DCNNs and meanwhile maintain both the numerical stability.

(3) All the DCNNs adopted in this study can achieve high accuracy in predicting the creep modulus of unseen microstructure, with the values of $R^2$ all above 0.96. Moreover, the correlation between creep modulus predicted by the DCNNs and microstructural parameters is consistent with that of the original database.

(4) The 4th consecutive convolution layer gives the feature maps that highlight the contribution of unhydrated cement particles and HD-CSH to the overall prediction of creep modulus, which correctly reflects the inherence of the microscale model.

## CRediT authorship contribution statement

**Minfei Liang:** Conceptualization, Methodology, Formal analysis, Investigation, Writing – original draft. **Yidong Gan:** Conceptualization, Methodology, Writing – review & editing, Supervision. **Ze Chang:** Methodology, Writing – review & editing, Visualization. **Zhi Wan:** Methodology, Writing – review & editing, Visualization. **Erik Schlangen:** Conceptualization, Resources, Writing – review & editing, Supervision. **Branko Šavija:** Conceptualization, Resources, Writing – review & editing, Supervision.

## Declaration of competing interest

The authors declare no conflict of interest.

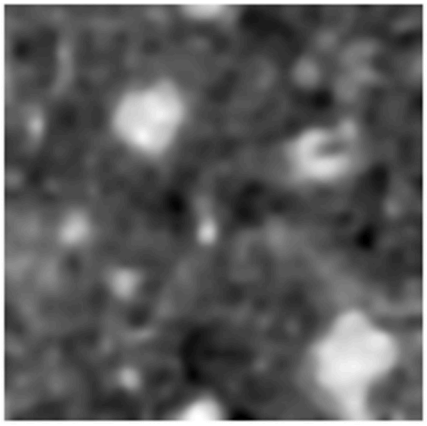
## Acknowledgements

Minfei Liang, Yidong Gan, Zhi Wan and Ze Chang would like to acknowledge the funding supported by China Scholarship Council under grant number 202007000027, 201706130140, 201906220205 and 201806060129.

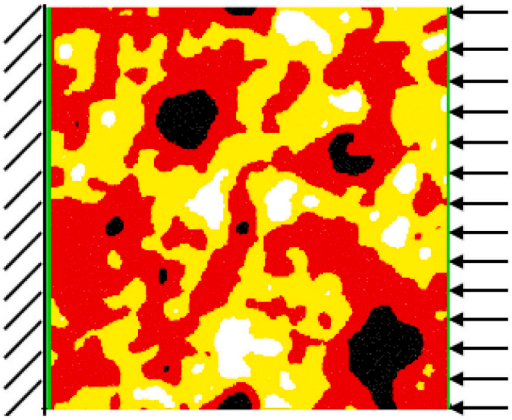
## Appendix A

### *A.1. Virtual specimens*

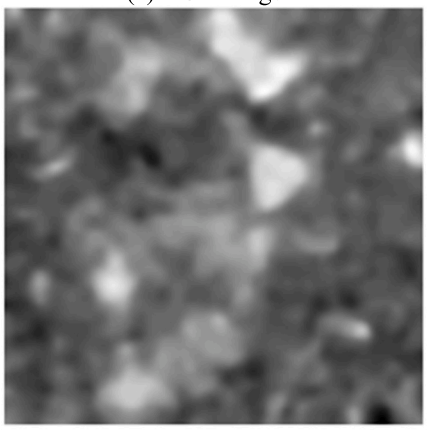
4 XCT scans and corresponding virtual specimens are randomly selected here to present the models, as is shown in Fig. A1. The green elements at the left-right boundaries are steel plate, whose modulus equal to 30,000 GPa.



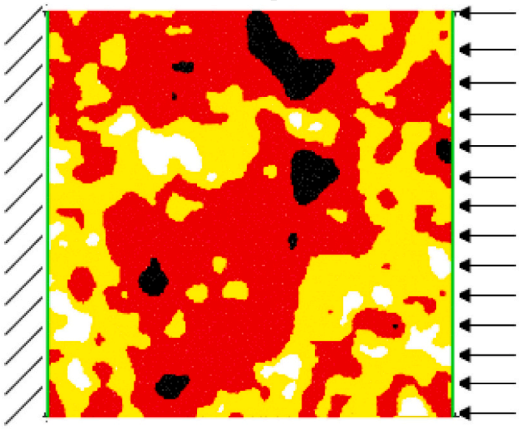
(a) XCT image 1



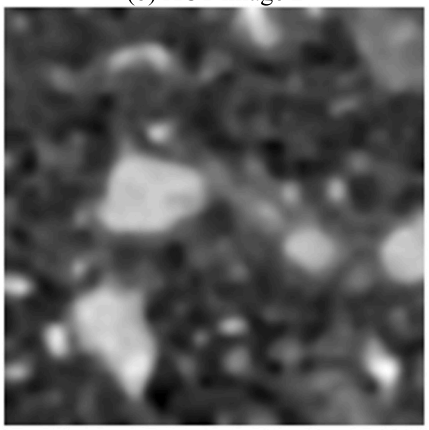
(e) Virtual specimen 1



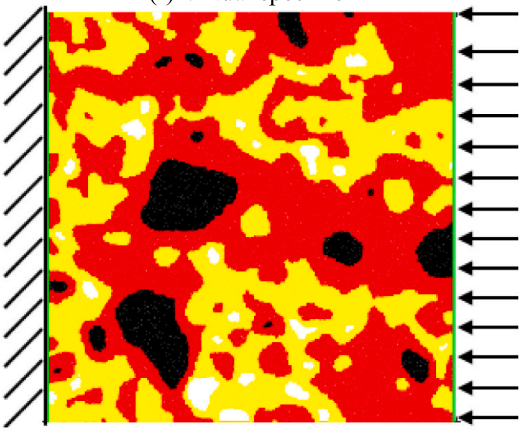
(b) XCT image 2



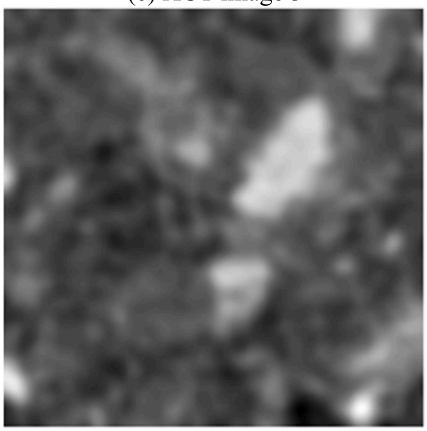
(f) Virtual specimen 2



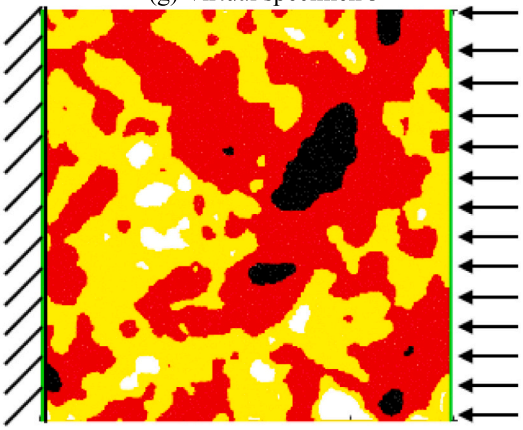
(c) XCT image 3



(g) Virtual specimen 3



(d) XCT image 4



(h) Virtual specimen 4

**Fig. A1.** Randomly selected virtual specimens
(Legend for different phases are consistent with Fig. 3).

### *A.2. Local stress distribution*

The local stress distribution of the four virtual specimens at time step $t = 30$s are shown in Fig. A2. As can be seen from the stress plot, the stress distribution is heterogenous depending on the microstructure, especially the distribution of pores: in the direction perpendicular to external load, stress concentration around the pores can be explicitly observed.

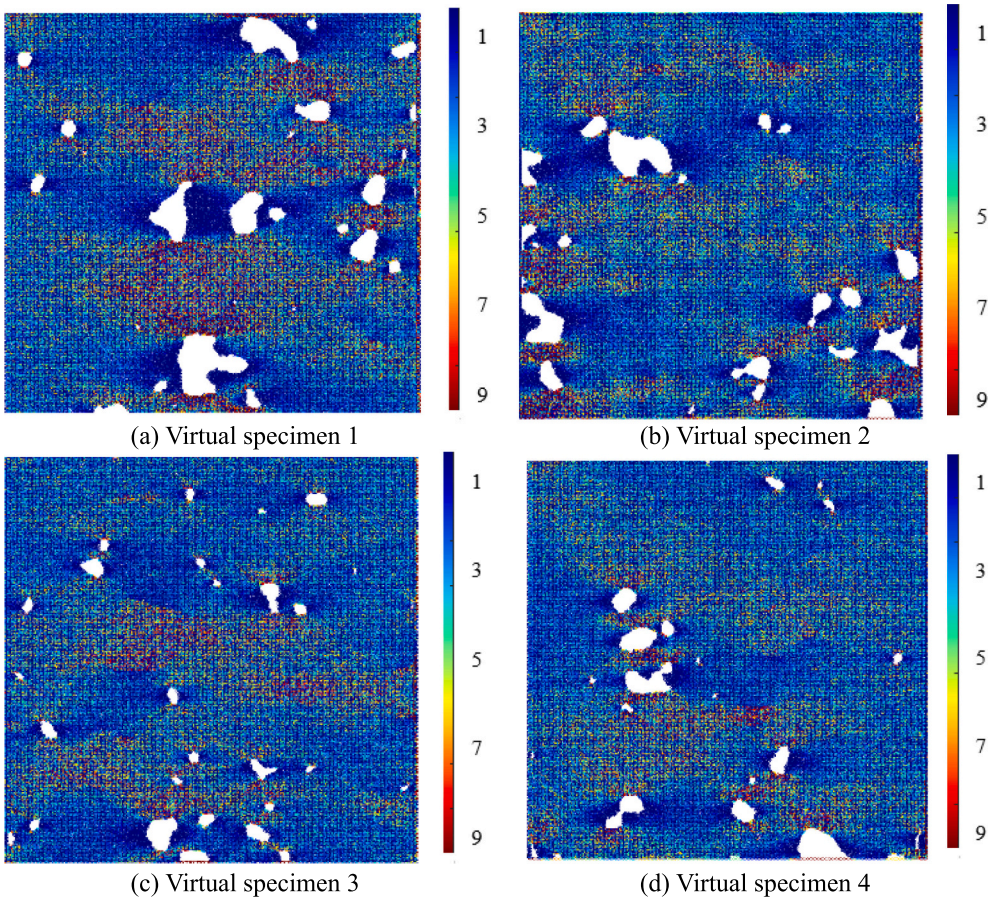



**Fig. A2.** Stress plot of 4 random selected virtual specimens (Unit: MPa).

### *A.3. Creep curves*

Based on the methods introduced in Section 2, the short-term creep behavior of a specific microstructure of cement paste can be calculated. For the 4 virtual specimens shown in Fig. A1, 3 main microstructural parameters including porosity, LD-CSH/ CSH ratio (LDR) and ratio of unhydrated cement particles (UHC) are shown in Table A1. Results of creep compliance of the 4 virtual specimens under 30 s' loading are shown in Fig. A3 and summarized in Table A1. The results in Fig. A3 are fitted by a creep compliance function with the form of Eq. (4). The exponent β remains constant for all specimens and equals to 0.4285 and the coefficient of determination are all higher than 0.995.

**Table A1**
Microstructural parameters of the 4 virtual specimens.

| NO. | *Porosity /%* | *LDR /%* | UHC /% | Creep modulus/ GPa |
|---|---|---|---|---|
| Virtual specimen 1 | 6.05 | 44.39 | 6.16 | 5614 |
| Virtual specimen 2 | 5.32 | 40.32 | 4.67 | 5796 |
| Virtual specimen 3 | 2.43 | 40.24 | 9.01 | 9361 |
| Virtual specimen 4 | 3.31 | 45.26 | 4.91 | 7059 |

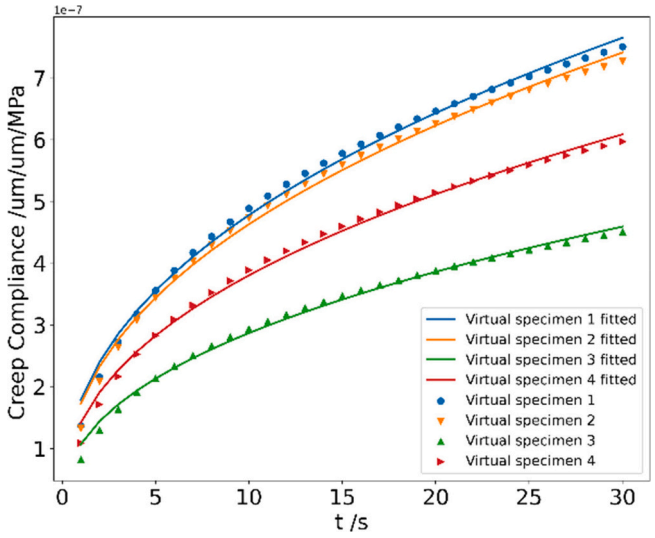



**Fig. A3.** Creep curves of 4 randomly selected virtual specimens.

## References


[1] F. Benboudjema, F. Meftah, J.-M. Torrenti, A viscoelastic approach for the assessment of the drying shrinkage behaviour of cementitious materials, Mater. Struct. 40 (2007) 163–174.

[2] S. Asamoto, K. Kato, T. Maki, Effect of creep induction at an early age on subsequent prestress loss and structural response of prestressed concrete beam, Constr. Build. Mater. 70 (2014) 158–164.

[3] Z.P. Bazant, L. Panula, Creep and shrinkage characterization for analyzing prestressed concrete structures, Prestress. Concr. Inst. 25 (1980) 86–122.

[4] Z.P. Bazant, M.H. Hubler, Q. Yu, Pervasiveness of excessive segmental bridge deflections: wake-up call for creep, ACI Struct. J. 108 (2011) 766–774.

[5] Dejian Shen, Yang Jiao, Yan Gao, Shuaishuai Zhu, Guoqing Jiang, Tensile creep and cracking resistance of concrete with different water-to-cement ratios at early age, Constr. Build. Mater. 146 (2017) 410–418.

[6] A. Kamen, E. Denarie, H. Sadouki, E. Bruhwiler, UHPFRC tensile creep at early age, Mater. Struct. 42 (2009) 113–122.

[7] Anja Estensen Klausen, Terje Kanstad, Øyvind Bjøntegaard, Erik Sellevold, Comparison of tensile and compressive creep of fly ash concretes in the hardening phase, Cem. Concr. Res. 95 (2017) 188–194.

[8] Gu. Chunping, Wang Yicong, Gao Fan, Yang Yang, Ni Tongyuan, Liu Jintao, Lou Xiaotian, Chen Jin, Early age tensile creep of high performance concrete containing mineral admixtures: experiments and modeling, Constr. Build. Mater. 197 (2019) 766–777.

[9] G. De Schutter, L. Taerwe, Fictitious degree of hydration method for the basic creep of early age concrete, Mater. Struct. 33 (2000) 370–380.

[10] Basile T. Tamtsia, James J. Beaudoin, Jacques Marchand, The early age short-term creep of hardening cement paste: load-induced hydration effects, Cem. Concr. Compos. 26 (2004) 481–489.

[11] Mateusz Wyrzykowski, Karen Scrivener, Pietro Lura, Basic creep of cement pastes at early age - the role of cement hydration, Cem. Concr. Res. 116 (2019) 191–201.

[12] Y. Theiner, M. Drexel, M. Neuner, G. Hofstetter, Comprehensive study of concrete creep, shrinkage, and water content evolution under sealed and drying conditions, Strain 53 (2017), e12223.

[13] Zhihui Zheng, Hu. Di, Peng Liu, Fei Sha, Lei Liu, Yu. Zhiwu, Considering the effect of the randomness of concrete strength and relative humidity on concrete creep, Struct. Concr. 1–15 (2020).

[14] Dejian Shen, Chengcai Li, Ci Liu, Ming Li, Jiacheng Kang, Experimental investigations on early-age tensile creep of internally cured high strength concrete under different initial stress/strength ratios, Constr. Build. Mater. 265 (2020), 120313.

[15] Su. Li, Yuan-feng Wang, Sheng-qi Mei, Peng-fei Li, Experimental investigation on the fundamental behavior of concrete creep, Constr. Build. Mater. 152 (2017) 250–258.

[16] Wei Jiang, Geert De Schutter, Yong Yuan, Degree of hydration based prediction of early age basic creep and creep recovery of blended concrete, Cem. Concr. Compos. 48 (2014) 83–90.

[17] Ya. Wei, Jingsong Huang, Siming Liang, Measurement and modeling concrete creep considering relative humidity effect, Mech. Time-Depend. Mater. 24 (2020) 161–177.

[18] Haidong Huang, Reyes Garcia, Shan-Shan Huang, Maurizio Guadagnini, Kypros Pilakoutas, A practical creep model for concrete elements under eccentric compression, Mater. Struct. 52 (2019) 119.

[19] Z.P. Bazant, S.T. Wu, Rate-type creep law of aging concrete based on Maxwell chain, Mater. Struct. 7 (1974) 45–60.

[20] Z.P. Bazant, Yunping xi. Continuous retardation Spectrum for solidification theory of concrete creep, J. Eng. Mech. 121 (2) (1995) 281–288.

[21] Giovanni Di Luzio, Luigi Cedolin, Carlo Beltrami, Tridimensional long-term finite element analysis of reinforced concrete structures with rate-type creep approach, Appl. Sci. 10 (2020) 4772.

[22] H. Do, S. Bishnoi, K.L. Scrivener, Microstructural modeling of early-age creep in hydrating cement paste, ASCE J. Eng. Mech. 142 (11) (2016), 04016086.

[23] Weinan Wang, Jinxin Gong, New relaxation function and age-adjusted effective modulus expressions for creep analysis of concrete structures, Eng. Struct. 188 (2019) 1–10.

[24] Y.-S. Park, Y.-H. Lee, Y. Lee, Description of concrete creep under time-varying stress using parallel creep curve, Adv. Mater. Sci. Eng. (2016) 13, https://doi.org/10.1155/2016/9370514, 9370514.

[25] A.M. Neville, Creep of concrete as a function of its cement paste content, Mag. Concr. Res. 16 (1964) 21–30.

[26] L. Granger, Comportement différé du béton dans les enceintes de centrales nucléaires: analyse et modélisation, Ecole Nationale des Ponts et Chaussees, 1995.

[27] Yue Li, Yunze Liu, Yinuo Li, Yaqiang Li, Rui Wang, Evaluation of concrete creep properties based on indentation test and multiscale homogenization method, Cem. Concr. Compos. 123 (2021) 10413.

[28] Ya. Wei, Siming Liang, Xiang Gao, Indentation creep of cementitious materials: experimental investigation from nano to micro length scales, Constr. Build. Mater. 143 (2017) 222–233.

[29] Hu. Zhangli, Mateusz Wyrzykowski, Michele Griffa, Karen Scrivener, Pietro Lura, Young's modulus and creep of calcium-silicate-hydrate compacts measured by microindentation, Cem. Concr. Res. 134 (2020), 106104.

[30] Yidong Gan, Matthieu Vandamme, Yu. Hongzhi Zhang, Erik Schlangen Chen, Klaas van Breugel, Branko Šavija, Micro-cantilever testing on the short-term creep behaviour of cement paste at micro-scale, Cem. Concr. Res. 134 (2020), 106105.

[31] Yidong Gan, Claudia Romero Rodriguez, Erik Schlangen, Klaas van Breugel, Branko Šavija, Assessing strain rate sensitivity of cement paste at the micro-scale through micro-cantilever testing, Cem. Concr. Compos. 121 (2021), 104084.

[32] Y Gan H Zhang B Šavija E Schlangen K Van Breugel. Static and fatigue tests on cementitious cantilever beams using nanoindenter. Micromachines 9 (12), 630.

[33] Y Gan H Zhang B Šavija E Schlangen K van Breugel. Micro-cantilever testing of cementitious materials under various loading conditions. Proceedings of the 10th International Conference on Fracture Mechanics of Concrete and Concrete Structures.

[34] Yidong Gan, Claudia Romero Rodriguez, Hongzhi Zhang, Erik Schlangen, Klaas van Breugel, Branko Savija, Modelling of microstructural effects on the creep of hardened cement paste using an experimentally-informed lattice model, Comput. Aided Civ. Inf. Eng. 35 (2020) 1085–1100.

[35] F. Lavergne, K. Sab, J. Sanahuja, M. Bornert, C. Toulemonde, Investigation of the effect of aggregates' morphology on concrete creep properties by numerical simulations, Cem. Concr. Res. 71 (2015) 14–28.

[36] Alain B. Giorla, Cyrille F. Dunant, Microstructural effects in the simulation of creep of concrete, Cem. Concr. Res. 105 (2018) 44–53.

[37] Petr Havlásek, Milan Jirásek, Multiscale modeling of drying shrinkage and creep of concrete, Cem. Concr. Res. 85 (2016) 55–74.

[38] Wei Jiang, Geert De Schutter, Yong Yuan, Degree of hydration based prediction of early age basic creep and creep recovery of blended concrete, Cem. Concr. Compos. 48 (2014) 83–90.

[39] Young-Jin Cha, Wooram Choi, Deep learning-based crack damage detection using convolutional neural networks, Comput. Aided Civ. Inf. Eng. 32 (2017) 361–378.
[40] Cao Vu Dung, Le Duc Anh, Autonomous concrete crack detection using deep fully convolutional neural network, Autom. Constr. 99 (2019) 52–58.
[41] S. Li, X. Zhao, Image-based concrete crack detection using convolutional neural network and exhaustive search technique, Adv. Civ. Eng. 2019 (2019) 12, https://doi.org/10.1155/2019/6520620, 6520620.
[42] Shengyuan Li, Xuefeng Zhao, Guangyi Zhou, Automatic pixel-level multiple damage detection of concrete structure using fully convolutional network, Comput. Aided Civ. Inf. Eng. 34 (2019) 616–634.
[43] Majdi Flah, Ahmed R. Suleiman, Moncef L. Nehdi, Classification and quantification of cracks in concrete structures using deep learning image-based techniques, Cem. Concr. Compos. 114 (2020), 103781.
[44] Jingwei Liu, Xu Yang, Stephen Lau, Xin Wang, Sang Luo, Vincent Cheng-Siong Lee, Ling Ding, Automated pavement crack detection and segmentation based on two-step convolutional neural network, Comput. Aided Civ. Inf. Eng. (2020) 1–15.
[45] Yupeng Ren, Jisheng Huang, Zhiyou Hong, Lu. Wei, Jun Yin, Lejun Zou, Xiaohua Shen, Image-based concrete crack detection in tunnels using deep fully convolutional networks, Constr. Build. Mater. 234 (2020), 117367.
[46] Xiao Liang, Image-based post-disaster inspection of reinforced concrete bridge systems using deep learning with bayesian optimization, Comput. Aided Civ. Inf. Eng. 34 (2019) 415–430.
[47] Mohsen Azimi, Gokhan Pekcan, Structural health monitoring using extremely compressed data through deep learning, Comput. Aided Civ. Inf. Eng. 35 (2020) 597–614.
[48] Shengyuan Li, Xuefeng Zhao, Guangyi Zhou, Automatic pixel-level multiple damage detection of concrete structure using fully convolutional network, Comput. Aided Civ. Inf. Eng. 34 (2019) 616–634.
[49] Byung Kwan Oh, Branko Glisic, Yousok Kim, Hyo Seon Park, Convolutional neural network-based wind-induced response estimation model for tall buildings, Comput. Aided Civ. Inf. Eng. 34 (2019) 843–858.
[50] Chengping Rao, Yang Liu, Three-dimensional convolutional neural network (3D-CNN) for heterogeneous material homogenization, Comput. Mater. Sci. 184 (2020), 109850.
[51] Charles Yang, Youngsoo Kim, Seunghwa Ryu, X.Gu. Grace, Prediction of composite microstructure stress-strain curves using convolutional neural networks, Mater. Des. 189 (2020), 108509.
[52] H.S. Wong, M.K. Head, N.R. Buenfeld, Pore segmentation of cement-based materials from backscattered electron images, Cem. Concr. Res. 36 (6) (2006) 1083–1090.
[53] 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 (6) (2000) 855–863.
[54] K. Van Breugel, Simulation of Hydration and Formation of Structure in Hardening Cement-based Materials, 1993. PhD Thesis.
[55] H. Zhang, B. Šavija, E. Schlangen, Towards understanding stochastic fracture performance of cement paste at micro length scale based on numerical simulation, Constr. Build. Mater. 183 (2018) 189–201.
[56] B. Šavija, M. Luković, E. Schlangen, Lattice modeling of rapid chloride migration in concrete, Cem. Concr. Res. 61–62 (2014) 49–63.
[57] Y. Huang, D. Yan, Z. Yang, G. Liu, 2D and 3D homogenization and fracture analysis of concrete based on in-situ X-ray computed tomography images and Monte Carlo simulations, Eng. Fract. Mech. 163 (2016) 37–54.
[58] M. Luković, E. Schlangen, G. Ye, Combined experimental and numerical study of fracture behaviour of cement paste at the microlevel, Cem. Concr. Res. 73 (2015) 123–135.
[59] A.B. Giorla, C.F. Dunant, Microstructural effects in the simulation of creep of concrete, Cem. Concr. Res. 105 (2018) 44–53.
[60] Z. Hu, A. Hilaire, J. Ston, M. Wyrzykowski, P. Lura, K. Scrivener, Intrinsic viscoelasticity of C-S-H assessed from basic creep of cement pastes, Cem. Concr. Res. 121 (2019) 11–20.
[61] C. Hu, Z. Li, A review on the mechanical properties of cement-based materials measured by nanoindentation, Constr. Build. Mater. 90 (2015) 80–90.
[62] M. Vandamme, F.J. Ulm, Nanoindentation investigation of creep properties of calcium silicate hydrates, Cem. Concr. Res. 52 (2013) 38–52.
[63] E. Schlangen, J.G.M. van Mier, Simple lattice model for numerical simulation of fracture of concrete materials and structures, Mater. Struct. 25 (9) (1992) 534–542.
[64] H. Zhang, B. Šavija, S.C. Figueiredo, E. Schlangen, Experimentally validated multi-scale modelling scheme of deformation and fracture of cement paste, Cem. Concr. Res. 102 (2017) 175–186.
[65] Z. Qian, Multiscale Modeling of Fracture Processes in Cementitious Materials, 2012. PhD Thesis.
[66] B. Šavija, H. Zhang, E. Schlangen, Micromechanical testing and modelling of blast furnace slag cement pastes, Constr. Build. Mater. 239 (2020), 117841.
[67] Ze Chang, et al., Lattice fracture model for concrete fracture revisited: calibration and validation, Appl. Sci. 10 (14) (2020) 4822.
[68] W. Lyu, E. Schlangen, K. van Breugel, Numerical Analysis of Effect of Micro-cracking and Selfhealing on the Long-term Creep of Cementitious Materials, RILEM, 2018.
[69] M. Königsberger, M. Irfan-ul-Hassan, B. Pichler, C. Hellmich, Downscaling based identification of nonaging power-law creep of cement hydrates, J. Eng. Mech. 142 (12) (2016) 1–11.
[70] Karen Simonyan, Andrew Zisserman, Very deep convolutional networks for large-scale image recognition, in: International Conference on Learning Representations (ICLR), 2015.
[71] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, ImageNet classification with deep convolutional neural networks, Commun. ACM 60 (6) (2017) 84–90.
[72] D. Scherer, A. Muller, S. Behnke, Evaluation of pooling operations in convolutional architectures for object recognition, in: Proceedings of the 20th International Conference on Artificial Neural Networks (ICANN), 2010.
[73] Aiman Al-Sabaawi, Hassan M. Ibrahim, Zinah Mohsin Arkah, Muthana Al Amidie, Laith Alzubaidi, Amended convolutional neural network with global average pooling for image classification, in: 20th International Conference on Intelligent Systems Design and Applications (ISDA), 2020.
[74] V. Nair, G.E. Hinton, Rectified linear units improve restricted boltzmann machines, in: Proceedings of the 27th International Conference on Machine Learning (ICML), 2010.
[75] Diederik P. Kingma, Jimmy Lei Ba, Adam: a method for stochastic optimization, in: International Conference on Learning Representations (ICLR), 2015.
[76] Matthew D. Zeiler, ADADELTA: An Adaptive Learning Rate Method, arXiv preprint arXiv, 2012, 1212.5701.
[77] Sinno Jialin Pan, Qiang Yang, A survey on transfer learning, IEEE Trans. Knowl. Data Eng. 22 (10) (2010) 1345–1359.