Engineering Applications of Artificial Intelligence 88 (2020) 103378

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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

Estimating cement compressive strength using three-dimensional microstructure images and deep belief network ☆

Jifeng Guo [a,1], Meihui Li [b,1], Lin Wang [a,*], Bo Yang [a,c], Liangliang Zhang [a], Zhenxiang Chen [a], Shiyuan Han [a], Laura Garcia-Hernandez [d], Ajith Abraham [e]

[a] Shandong Provincial Key Laboratory of Network based Intelligent Computing, University of Jinan, Jinan, 250022, China
[b] Shandong Provincial Key Laboratory of Oral Tissue Regeneration, Department of Human Microbiome, School of Stomatology, Shandong University, Jinan, Shandong 250012, China
[c] School of Information Science and Engineering, Linyi University, Linyi, 276000, China
[d] Area of Project Engineering, University of Cordoba, Spain
[e] Machine Intelligence Research Labs (MIR Labs), Scientific Network for Innovation and Research Excellence, Auburn, USA

ARTICLE INFO

Keywords:
Cement compressive strength
Microstructure images
Deep belief network
Graphics processing units

ABSTRACT

The estimation of cement compressive strength is of great significance in the quality inspections, technological designs, and engineering applications for cement. Compared to destructive methods, the nondestructive estimation approaches save the cost in the manpower and material. However, the existing nondestructive methods have the large error because the used influence factors are difficult to control and the used two-dimensional microstructure images can not reflect the specific spatial structure of the entire cement. In this paper, a novel model is proposed to estimate the cement compressive strength using three-dimensional microstructure images and deep belief network. To reduce the computation consumption induced by three-dimensional images with abundant information, this method extracts image features that reflect the cement hydration state to estimate cement compressive strength. Deep belief network is applied to build the estimation model. Its unique training pattern and flexibility of parameters improve the ability to learn nonlinear relationships between microstructure images and cement compressive strength. Furthermore, the training processes are accelerated on the graphics processing units. The experimental results prove that the proposed method estimates cement compressive strength nondestructively and improves the efficiency.

1. Introduction

Cement is widely used in numerous fields including construction and hydraulic engineering, and its quality greatly impacts the safety of production and life. Strength is one main criterion to measure the cement quality (Wang et al., 2016a) and cement compressive strength (CCS) is the most important factor (de Siqueira Tango, 1998). Thus, estimating CCS is conducive to guide the application of cement in the production and the life. Moreover, its nondestructive estimation helps to save manpower and material.

Estimating cement compressive strength is multivariable and nonlinear. The CCS is affected by many factors, including water-to-cement ratio (w/c), age (Baykasoğlu et al., 2004), clinker composition (Tsivilis and Parissakis, 1995), chemical composition (Zhang and Napier-Munn, 1995), and fly ash fineness (Chindaprasirt et al., 2005). Many scholars have investigated cement compressive strength using those factors and several kinds of methods (Gao, 1997; Sebastiá et al., 2003). However, since measuring macroscopic materials exists error and environmental variables are volatile, these measurements cannot accurately reflect hydration state of cement. Therefore, the estimation results have a large errors. Although two-dimensional (2-D) microstructure images were used afterward (Li et al., 2016, 2017), the results were still unsatisfactory because 2-D images can not reflect the specific spatial structure of entire cement sample. Neural networks outperform other intelligent methods in estimating CCS. However, the ability of the shallow neural network to represent complex functions is limited for its architectures (Bengio et al., 2007), which prevent the error from being further reduced. Deep neural networks have better approximation ability than shallow ones with the same number of parameters (Bianchini and Scarselli, 2014; Seide et al., 2011). Its multiple hidden layer structure has better feature learning and approximation ability, which benefits prediction and classification (Hinton and Salakhutdinov, 2006).

☆ No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.engappai.2019.103378.

* Corresponding author.
E-mail address: wangplanet@gmail.com (L. Wang).

[1] Jifeng Guo and Meihui Li contributed equally to this study.

https://doi.org/10.1016/j.engappai.2019.103378
Received 1 March 2019; Received in revised form 11 November 2019; Accepted 12 November 2019
Available online 20 November 2019
0952-1976/

Considering the fact that three-dimensional (3-D) microstructure images reflect hydration state accurately and contain the spatial structure of the different phase, this study uses 3-D microstructure image to estimate the CCS. Despite the abundant information, it increases the computational consumption. In order to solve this problem, the image features that reflect the cement hydration state are extracted. In addition, deep belief network (DBN) (Kuremoto et al., 2014) demonstrates outstanding performance in various problem and has the ability to discover the mysterious between microstructural images and CCS. It is responsible for modeling. However, its training processes are high time-consumption. The graphics processing units (GPUs) (Larhlimi and Mestari, 2018) environment is introduced to accelerate these processes.

This study proposes a novel method for estimating CCS using 3-D microstructure images and DBN. The contributions of this paper are elaborated in two aspects.

- *It is the first time to estimate the cement compressive strength by using three-dimensional microstructure image.*
- *It is the first time to estimate the cement compressive strength by using deep belief network, and it is parallelized on the GPUs.*

The structure of this paper is as follows: Section 2 introduces the related works on the estimation of CCS, deep neural networks, and compute unified device architecture (CUDA) programming. Section 3 elaborates the motivation of this research. Section 4 introduces the steps of the method in detail. Section 5 presents the experiments and the comparison of the results. Finally, Section 6 concludes full paper.

2. Related works

2.1. The estimation of cement compressive strength

Estimating cement compressive strength is a multivariable and nonlinear (Zhang et al., 2018) problem. Existing studies applied various nonlinear methods to estimate CCS according to the hydration ratio of the materials and other influencing factors. Tsivilis and Parissakis (1995) applied regression analysis to build a model for predicting compressive strength of Portland cement after 2, 7 and 28 days, showing the importance of chemical-mineralogical and fineness factors. de Siqueira Tango (1998) presented a strength-time function to predict CCS of a late age, proving that this function can predict strengths satisfactorily only needing to know the type of cement. Gao (1997) analyzed the fuzzy logic method and applied it to predict CCS on the various samples. Akkurt et al. (2004) also applied fuzzy logic to predict the 28-day compressive strength of cement mortar. In addition, artificial neural network (ANN) (Yeh, 1998; Sebastiá et al., 2003), gene expression programming (Baykasoğlu et al., 2004; Guo et al., 2016), multi-layer feed-forward neural networks (Ni and Wang, 2000), and multi-expression programming (Zhang et al., 2012) are also applied to predict CCS according many influence factors. Particularly, the neural network structure like a brain network (Wang and Orchard, 2018) and has strong fault tolerance (Han et al., 2017), achieving good results. However, it has a shock phenomenon. Optimizing its weights by Back-propagation (BP) (Ni and Wang, 2000), genetic algorithm (Sangdani et al., 2018), and other optimization methods (Wang et al., 2016b; Chen et al., 2016) makes it shows superior performance. In addition, other types of neural networks have also been proposed (Huang et al., 2017).

Actually, the prediction results using influence factors have large error. Moreover, the difficulty exists in measuring the hydration ratio of the materials and micro components. Therefore, many scholars studied the strength via cement images. Li (2004) used backscattering and X-ray images to predict CCS. However, it is only a rough estimation of the extracted phase and cannot completely reflect the hydration state of the material. Therefore, this method has a prediction error. Doğan et al. (2015) obtained digital images of the surface of concrete cube specimens by the digital camera. After image processing, gray-level histogram is extracted to estimate CCS. The acquisition of gray images that reflect the hydration state is limited by the influence of illumination. Moreover, those gray images fail to completely reflect the structure of the sample, which result in estimation errors. Li et al. (2016) estimated the CCS using two-dimensional cement images and neural networks. This method extracts features from the gray-level histogram (GLH) and gray level co-occurrence matrix (GLCM) of cement images and proves that using the cement microstructure images to estimate the CCS is feasible. Later, convolutional neural networks were also applied to estimate the strength using two-dimensional images. Compared with using traditional neural network, this work reduced the estimation error (Li et al., 2017). However, the 2-D microstructure image is difficult to reflect the specific spatial structure of the entire cement sample.

2.2. Deep neural networks

Neural networks enable approximating any nonlinear functions, becoming the most available algorithms in the field of machine learning (Bao et al., 2019) and have developed to multilayer network architectures (Zhang et al., 2019b). Compared with shallow architectures, deep architectures enable mitigating oscillation (Bohner et al., 2018; Chatzarakis and Li, 2018) and show superior performance in the feature extraction and modeling (Chen et al., 2018; Tao et al., 2018) with a large number of parameters. Particularly, DBN (Hinton, 2009) was proposed in recent years and was widely applied to various tasks (Luo et al., 2016). Ren and Wu (2014) learned the feature of electroencephalographic data by convolutional deep belief networks and applied it to many datasets from brain-computer interfaces competitions, discovering this method have better performance. Xie et al. (2014) extracted high-level features automatically from High Speed Train (HST) vibration signals using DBN and the experimental results show that it helps to diagnose the different faults of HST. Mohamed et al. (2011) discovered deep belief network enable extracting the multiple layers of features that capture the higher-order statistical structure of the data. They applied it to build a model that had already been used successfully for phone recognition, realizing good performance. Yin and Zhao (2016) proposed an automated diagnosis network of vehicle on-board equipments for high-speed train via deep belief network. The results show that the developed DBN outperforms both k-nearest neighbor and ANN-BP, improving the accuracy to 90∼95%. Son et al. (2018) proposed an artifact elimination model for pulse waveform signals based on DBN, unearthing DBN exhibited higher sensitivity than SAI in identifying artifacts. This model enhanced the quality of signal analysis and had applied to the artifact removal in monitoring arterial blood pressure. A large number of studies (Li et al., 2019; Jang et al., 2017) have shown that DBN presents better performance. However, to obtain efficient neural networks and achieve the high performance, its training creates a high consumption in both time and calculation (Peddabachigari et al., 2007).

2.3. Compute unified device architecture

Compute unified device architecture (Xavier et al., 2010) runs on a graphics processing unit and it enables saving the calculated time. The CUDA considers GPU as a computing device that manages and distributes parallel data (Wang et al., 2014b). CPU is regarded as a processing device for serial operations. The CUDA programming combines the serial CPU side with the parallel GPU side to improve training efficiency. Recently, general purpose GPU is applied not only in graphics processing but also in intensive computing tasks.

Wang et al. (2018) proposed a parallel nearest neighbor partitioning neural network classifier based on compute unified device architecture. It allocates blocks and threads reasonably to evaluate neural networks and to perform parallel subtasks. The experimental results manifest that the proposed method can achieve shortening of training time on several datasets, especially the speedup ratio on the MAGIC data set

is up to 164.29 times. van der Laan et al. (2011) accelerated Discrete Wavelet Transform using CUDA, which significantly reduces the memory usage and time consumption. Rovenskaya and Croce (2015) solved the Boltzmann equation on GPUs. Its processing speed increased by 50 times. Beyeler et al. (2015) presented a GPU-accelerated cortical neural network model for visually guided robot navigation. Gong et al. (2017) proposed a novel immune convolutional neural network algorithm and accelerated it on the NVIDIA GPU. Compared with the traditional convolutional neural network, this method is excellent in recognition rate, performance stability, and computing speed. Khomenko et al. (2016) used sequence bucketing and multi-GPU data parallelization to accelerate recurrent neural network training and the processing time is shortened.

3. Motivation

Cement compressive strength not only is related to the hydration ratio of the materials but is affected by the environment. Therefore, a certain error exists between the estimated and actual value. The accurate estimation of CCS will enable nondestructive estimation of strength and save a lot of manpower and material. At present, many scholars have conducted considerable research on the CCS according to the influencing factors and employing various methods. However, the factors that affect CCS are not completely controllable and cement hydration products are difficult to measure, which make large estimation error in the results. Although two-dimensional images were applied to predict CCS afterward, it cannot reflect the specific spatial structure of the entire cement sample, resulting in unsatisfactory prediction results.

Three-dimensional microstructure images reflect hydration state accurately and contain the spatial structure of the different phase, which motivate us to use 3-D microstructure images to estimate the CCS. Despite the abundant information, it increases the resource consumption. In order to improve efficiency, image features that reflect the state of the cement are extracted. Furthermore, having the excellent performance, deep learning is applied to build a CCS estimation model. Although convolutional neural networks are expert in image recognition, its convolution operation has a poor ability to recognize fuzzy boundaries. Therefore, it is not suitable for identifying cement images that contain many substances which are difficult to distinguish boundaries. DBN enables to extract high-level features contained in the sample. Moreover, its flexibility of parameters and unique training pattern improve the ability to learn nonlinear relationships between cement images and CCS. Thus this paper adopts DBN. In order to overcome high time consumption, its training processes are accelerated on the compute unified device architecture.

4. Methodology

The main flow chart of the proposed method is shown in Fig. 1. Firstly, features are extracted from the 3-D microstructure images and the corresponding strength are recorded. The features extracted from the gray-level histogram and gray level co-occurrence matrixes are the statistics features, which reflect the phase state and texture information in cement hydration. Secondly, the DBN are adopted to analyze the input features, and to learn the relations between features and strength. Its training process includes high-level features extraction in restricted Boltzmann machine (RBM) and weight adjustment. However, its training process suffers from high time consumption. In order to improve efficiency, it is accelerated on CUDA platform.

4.1. Feature extraction

The 3-D microstructure images of cement which are obtained by Micro-CT (Wang et al., 2014a) display a more accurate gray-value than that obtained by CT (Chen et al., 2017). The linear attenuation coefficient of different substances corresponds to the different gray-values in images. Therefore, its different gray-values represent different substances during the hydration. In addition, since the microstructure and physical properties have a dynamic relationship (Wang et al., 2015; Zhang et al., 2019a), 3-D microstructure images are able to reflect the properties of cement and is suitable for CCS estimation.

The histogram enables describing the images, but it only reflects the number of micro-substances, i.e., it is difficult to represent the spatial structure of micro-substances. Therefore, the general distribution of micro-substances during cement hydration is described by the GLH of 3-D cement microstructure images, and the spatial relationship that the 3-D cement microstructure images reflect is represented by the GLCM. Since they can reflect the cement hydration state, the GLH and GLCM are obtained from the 3-D microstructure gray images, and the extracted features are described in Fig. 2. The selection of features and its process will be described in detail below.

4.1.1. Gray-level histogram

The GLH describes the global distribution of gray-values in 3-D microstructure images, i.e., the GLH indicates the proportion of different gray-values in the entire 3-D microstructure image (Shi et al., 2007). The height of the gray value directly reflects the amount of material because each gray-value corresponds to a substance or its combination, i.e., the GLH can describe the distribution of materials in the microstructure. Furthermore, the GLH can be considered a discrete function expressed as Eq. (1) (Murat et al., 2012).

$$H(g) = \frac{n_g}{N}, g = 0, \ldots, L-1 \tag{1}$$

where g is the gray-level in the 3-D microstructure images, and L is the total number of the gray-levels, n_g is the quantity of the gray-level g, and N is the total number of image pixels. Moreover, N is also the sum of the n_g, i.e. they satisfy the relationship $N = \sum_{g=0}^{L-1} n_g$. The GLH is usually described by several features. These features are introduced as follows.

Mean (μ) expresses the mean value of the gray-level in the 3-D microstructure images. The mean also reflects the average content of the hydrated products, the unhydrated particles, air voids, and the pores in the cement hydration. It is expressed by Eq. (2)

$$\mu = \sum_{g=0}^{L-1} g \times H(g) \tag{2}$$

Variance (σ^2) represents the discrete degree of the gray-level in the 3-D microstructure images. Furthermore, it reflects the content relationship between the hydrated products, air voids, the unhydrated particles, and the pores in the cement hydration. It is inversely proportional to the content of the hydration products and is expressed by Eq. (3)

$$\sigma^2 = \sum_{g=0}^{L-1} (g-\mu)^2 \times H(g) \tag{3}$$

where σ is the standard deviation of the GLH.

Skewness (μ_s) evaluates the asymmetry degree of the gray-level in distribution and is expressed by Eq. (4). The content of hydrated products is inversely proportional to it, while the content of the unhydrated particles, and pores are directly proportional to it.

$$\mu_s = \frac{1}{\sigma^3} \sum_{g=0}^{L-1} (g-\mu)^3 \times H(g) \tag{4}$$

Kurtosis (μ_k) evaluates the distribution condition which focuses on the near of the mean gray-level. Moreover, it verifies whether the hydration state is complete and is directly proportional to the degree of hydration. It is expressed by Eq. (5).

$$\mu_k = \frac{1}{\sigma^4} \sum_{g=0}^{L-1} (g-\mu)^4 \times H(g) - 3 \tag{5}$$

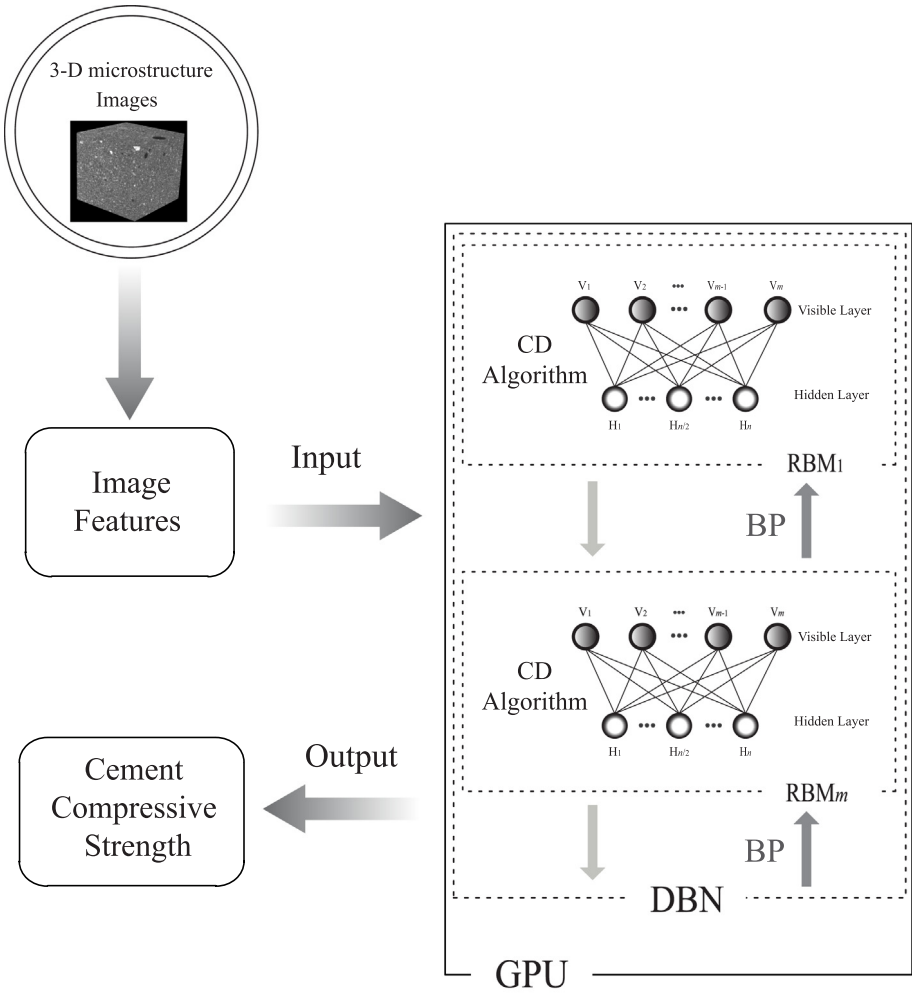

Fig. 1. Main flow chart of the estimating CCS using 3-D microstructure images features and DBN based on GPUs.

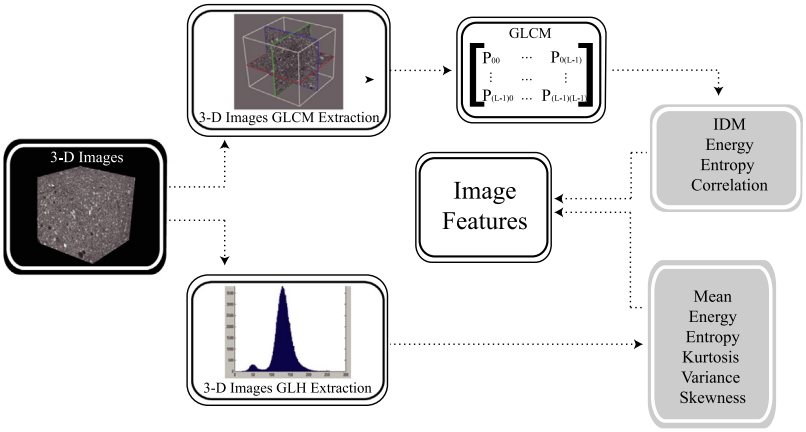

Fig. 2. Main process of feature extraction and the extracted features.

Energy (μ_n) shows the homogeneous degree of the gray-level distribution. When the hydrated products are more uniform, the energy is higher. It is expressed by Eq. (6).

$$\mu_n = \sum_{g=0}^{L-1} H(g)^2 \tag{6}$$

Entropy (μ_i) describes the uniformity of the gray-level distribution and is inversely proportional to the distribution of the hydrated products, air voids, unhydrated particles, and pores. It is expressed by Eq. (7).

$$\mu_i = \sum_{g=0}^{L-1} H(g)\log_2[H(g)] \tag{7}$$

4.1.2. Gray level co-occurrence matrix

The texture characteristics of 3-D cement microstructure images reflect the characteristics of the particle distribution and internal structure of the cement. Therefore, the texture features of 3-D images can describe the inherent regular pattern of the microstructure. The texture of the images is represented by the gray-level distribution. In 1973, Haralick et al. (1973) proposed the concept of GLCM. It reflects the gray value and the gray-level distribution of the 3-D microstructure images (Hu et al., 2011) and also describes the spatial distribution and spatial correlation of the gray-level (Wang and Kong, 2011). Moreover, the GLCM of the 3-D microstructure images can reflects the integrated information about the direction, interval, and variation range.

The GLCM is defined through the joint probability density of two pixels (He et al., 2013) and presents the statistics of the probability from two pixels with the gray values of i and j simultaneously, which denote the straight-line distance d and the angle of θ, respectively. The GLCM is expressed as Eq. (8) (Malegori et al., 2016).

$$p(i,j,d,\theta) = \#\{[(x_i,y_i),(x_j,y_j)] | f(x_i,y_i) = i, f(x_j,y_j) = j, d, \theta\} \tag{8}$$

where i and j are the gray values of two pixels, d is the distance, and θ is the angle. $\#$ represents the elements number of the set. x is the abscissa, y is the ordinate of the pixel (x, y), and $f(x, y)$ is the gray value of the pixel.

The second-degree statistics features is extracted from the GLCM to avoid a massive calculation. The GLCM has 14 different features in total, and those considered features in this study are as follows Malegori et al. (2016).

Entropy (ENT) reflects the non-uniform degree or complexity of the texture in 3-D images. A large value expresses a large complexity of the texture and a large complexity of the spatial structure of the hydrated products and the unhydrated particles. It is expressed by Eq. (9).

$$ENT = -\sum_{i=0}^{L-1}\sum_{j=0}^{L-1} P(i,j)\log P(i,j) \tag{9}$$

where $P(i, j)$ is the elements of the GLCM, and the variables i and j are independent of each other.

Correlation (COR) expresses the consistency and correlation of the local gray level in the 3-D microstructure images. A large value indicates that the distribution of concentration from the same hydrated products in the hydration is higher. It is expressed by Eq. (10).

$$COR = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1} \frac{ijP(i,j) - \mu_x\mu_y}{\sigma_x\sigma_y} \tag{10}$$

where μ_x is the average gray-level and μ_y is the mean smoothness. σ_x^2 and σ_y^2 is the variance of gray-level and smoothness, respectively.

Inverse different moment (IDM) measures the mutative degree of local texture. Its value is directly proportional to the uniformity of the local texture in 3-D images. In other words, the IDM indicates whether the local spatial structure of all kinds of hydrates is uniform at the hydration state. It is expressed by Eq. (11).

$$IDM = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1} \frac{P(i,j)}{1+(i-j)^2} \tag{11}$$

Energy (ENG) describes the degree of uniformity of the gray-level distribution and the thickness of the texture. A large value shows that the structure of the substance distribution is uniform and the hydration products are regular. It is expressed by Eq. (12).

$$ENG = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1} \{P(i,j)\}^2 \tag{12}$$

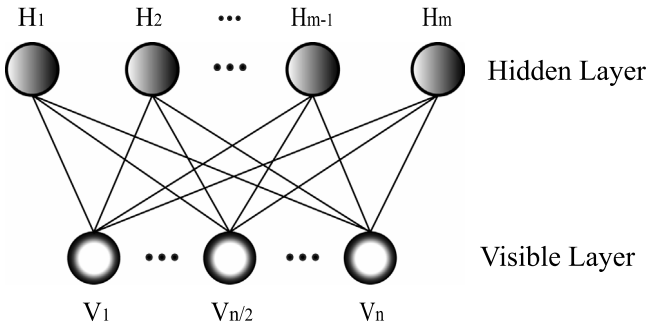

Fig. 3. The structure of RBM, which including n visible neurons and m hidden neurons. Moreover, the RBM is full connection between visible and hidden neurons.

4.1.3. Extraction process

The process of extracting features from cement images are as follows:

(1) Calculating the gray-level histogram of 3D image and extracting the selected features according to Section 4.1.1;
(2) Calculating GLCM at different direction in each distance and extracting its features, including mean and standard deviation of ENT, COR, IDM, and ENG at different directions.

The features of data consist of the features obtained from the gray-level histogram and the gray level co-occurrence matrixes.

4.2. Estimation of cement compressive strength

After feature extraction, the DBN-based model is trained on cement data, which contains features and CCS. Model's input is the extracted feature and its output is the estimated CCS. In training, it has two main processes that ensure high performance. The first one is extracting abstract features using multi-layer RBMs and the second one is tuning weights through the BP algorithm. Moreover, the training objective function (Zhou et al., 2016) is mean square error and the ultimate goal is minimizing its value.

Multi-layer RBM (Hinton and Sejnowski, 1986) in DBN extract abstract features and its structure is shown in Fig. 3. The RBM consists of some visible and hidden neurons. The connections are undirected and only exist between visible and hidden layer (Sarikaya et al., 2014). The training algorithm used in RBM is contrastive divergence (CD) algorithm (Sheri et al., 2015) and the more abstract features are extracted from input data after RBM training. Through multi-layer training, the features of data are more effective. Since the time consumption of this process is high, it is accelerated on the GPU, and the acceleration is described in detail in the next subsection.

After the pre-training process of the DBN, the higher-level features of the data are extracted. Meantime, the connecting weights between the networks are initialized, which is different from those chaotic and stochastic patterns in traditional neural network (Shang et al., 2019). In order to obtain better prediction results, the entire network also needs "fine tuning" via BP algorithms to adjust weights. However, the training process is time-consuming because of a large number of samples and long training time of the depth neural network (Chen and Liu, 2018). In order to reduce training time, it is accelerated on the GPU.

4.3. GPU acceleration

The CUDA is helpful to solve the time-consuming problem in deep structure and learning process. Its main advantage is that it treats GPU as a parallel machine with a single instruction and multiple threads (SIMT). The structure of the GPU is shown in Fig. 4. Each GPU has

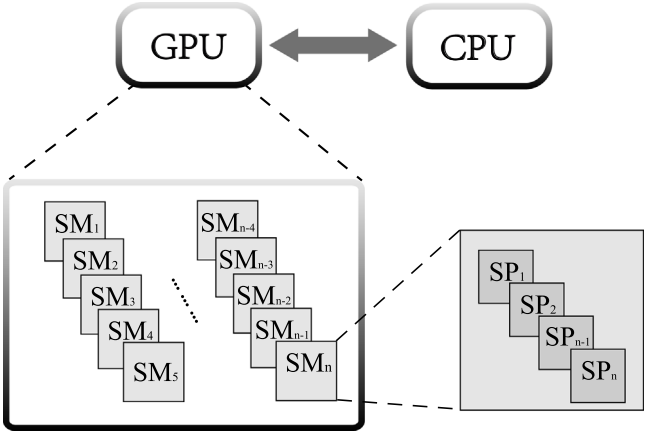

Fig. 4. The structure of GPU. One GPU consists of several SMs, and each SM consists of several SPs.

several streaming multiprocessor (SM) (Rios et al., 2018), and each SM has many streaming processors (SP) (Lopes and Ribeiro, 2014). The data are transmitted to the global memory of GPU from the CPU. Moreover, the shared memory in the GPU enhances the efficiency of data transmission. Within the allowable number of threads, the big number of parallel threads correspond to high efficiency.

During the training process of DBN, a large number of correlations exist between the former and later data. For simple operations, when all the steps are parallelized, its time will increase and efficiency will decrease. Therefore, numerous steps can only be implemented in series. The CPU terminal controls the entire serial processes of the RBM and BP algorithm, while the GPU terminal controls the parallel implementation of the subprocess (Wang et al., 2010). The data transmissions between the GPU and CPU are time-consuming. In order to improve the computational efficiency, the weights and biases of the DBN, the image features, and strength values are transmitted to the GPU before the calculation. Fig. 5 depicts parallel implementation of two processes. Images features input the RBMs to obtain the weights and abstract features of cement microstructure. After fine-tuning by BP algorithm, the new output is connected to the output layer to estimate the CCS.

The training process of each RBM consists of three steps, namely, the mapping of the visible layer to hidden layer, the mapping of the hidden layer to visible layer, and the mapping of the visible layer to hidden layer. The CPU controls three steps of the CD algorithm in the serial computing, while the GPU controls every subprocess in parallel computing. In order to achieve the maximization of parallelism, algorithm must to make each SM and SP fully used according to the architecture of the GPU. One step of the CD algorithm calculates the abstract features of the hidden layer. Firstly, the weights and biases of RBM are stored in the global memory. Moreover, the features of one batch are included in the array v_1 and also transmitted to the global memory. Secondly, one thread calculates one feature mapping h_1, v_2 and h_2. The number of thread is n_{thread}. Each thread separately calculates the value of neurons in each layer. Each block is divided into two dimensions x and y. The neurons of each layer is x dimension. According to the maximum number of threads in one block, the y dimension is decided. The number of thread in x dimension is x_{thread}. The number of the block are distributed according to the samples in parallel. Finally, the weights and biases are updated under the results of the previous steps.

Through the CD algorithm of several RBMs, the weights and the abstract features of cement microstructure are obtained. Next, the weights and biases of the DBN are fine-tuned with BP algorithm to reduce the error. The BP algorithm includes forward and backward propagation. The batch of samples in parallel are also used to adjust the weights in DBN for achieving a more accurate estimation. Firstly, the parallel strategy of forwarding propagation is the same as the CD algorithm. The second part is calculating the gradient of the weights and biases in parallel according to the gradient descent method. The last part is updating the weights and biases.

Table 1
Chemical compositions and physical properties of cement.

Type	Cement A	Cement B	Cement C
CaO (%)	63.89	63.74	64.74
SiO_2 (%)	22.58	20.93	20.72
Fe_2O_3 (%)	3.03	3.77	3.74
Al_2O_3 (%)	4.67	4.45	4.66
AgO (%)	2.46	2.51	3.34
SO_3 (%)	1.75	1.56	1.38
K_2O (%)	1.2	0.93	0.54
Na_2O (%)	0.17	0.23	0.19
$fCaO$ (%)	0.64	1.01	1.25
C_3S (%)	45.12	56.5	60.34
C_2S (%)	30.79	17.47	13.97
C_3A (%)	7.52	5.42	6.03
C_4AF (%)	9.21	11.46	11.37
$Specific\ surface\ area$ (cm^2/g)	4088	3475	4397
$Sieve\ residue\ on\ 74\ \mu$ (%)	3.56	6.44	3.63

5. Experiments and results

In order to ensure that the selected cement samples are represented sufficiently, three types of cement are adopted and labeled as A, B, and C according their chemical compositions and physical. Table 1 shows the specific information about three types of cement. The cement A has high dicalcium silicate (C2S), which contributes to the enhancement of strength at the later age. However, its hydration rate is low at the early ages. The cement C has high tricalcium silicate (C3S), which accelerates the hydration. The cement B is the ordinary one, and its hydration rate is between the other two types of cement.

In order to obtain the image data, the cement powder is fully mixed with water at first. And then, the paste is injected into molds by the injector, and the needle's diameter is 1.6 mm to ensure that the w/c does not be changed. The cement paste is cured in a specific environment (20 °C and 95% humidity) for 24 h. After removing the mold, cement cylinder (the diameter is 4 mm) can be obtained. SkyScan 1172 High-Resolution desktop Micro-CT System scans cement cylinder at the 2nd, 3rd, 4th, 5th, 6th, 7th, 14th, 21st, and 28th days (Wang et al., 2014a). After a series of image processing, including 3-D reconstruction, artifact removal, and 3-D registration, the obtained cross-section image is reconstructed into three-dimensional microstructure images. The 3-D microstructure images on the 2nd day of each type of cement are shown in Fig. 6. In addition, since 3-D microstructure images with size of $70 \times 70 \times 70$ can cover all phases of cement, the images with this size are selected as one cement sample. Each type of cement has 9000 samples and the total data set contains 27,000 samples. The specific values of strength are obtained utilizing the compressive strength tester.

The selected features are extracted from GLH and GLCM of 3-D images. When calculating the GLCM, the distance is set to 1, 2, 3, 4, 5, and the number of direction is set to 13. In this work, 46 features are obtained from the 3-D microstructure images, in which 6 features (Mean, Variance, Skewness, Kurtosis, Energy and Entropy) are derived from the GLH describing the distribution of materials, and 40 features that reflect cement textures are obtained from the GLCM in 5 distances. At each distance, the GLCM has 8 features, which are the Mean(μ) and Standard Deviation(σ) of four features (ENT, COR, IDM and ENG) about GLCM at 13 directions. All extracted features of 3-D microstructure images are shown in Table 2. The 1st to 6th rows are the features from the GLH. In addition, the other features from the GLCM are divided into 5 groups by different distances, each of which represents the μ and σ of ENG, ENT, IDM and COR at a certain distance, respectively.

In this paper, three types of cement samples are trained and tested. For each type of cement sample, 8500 samples are trained and 500

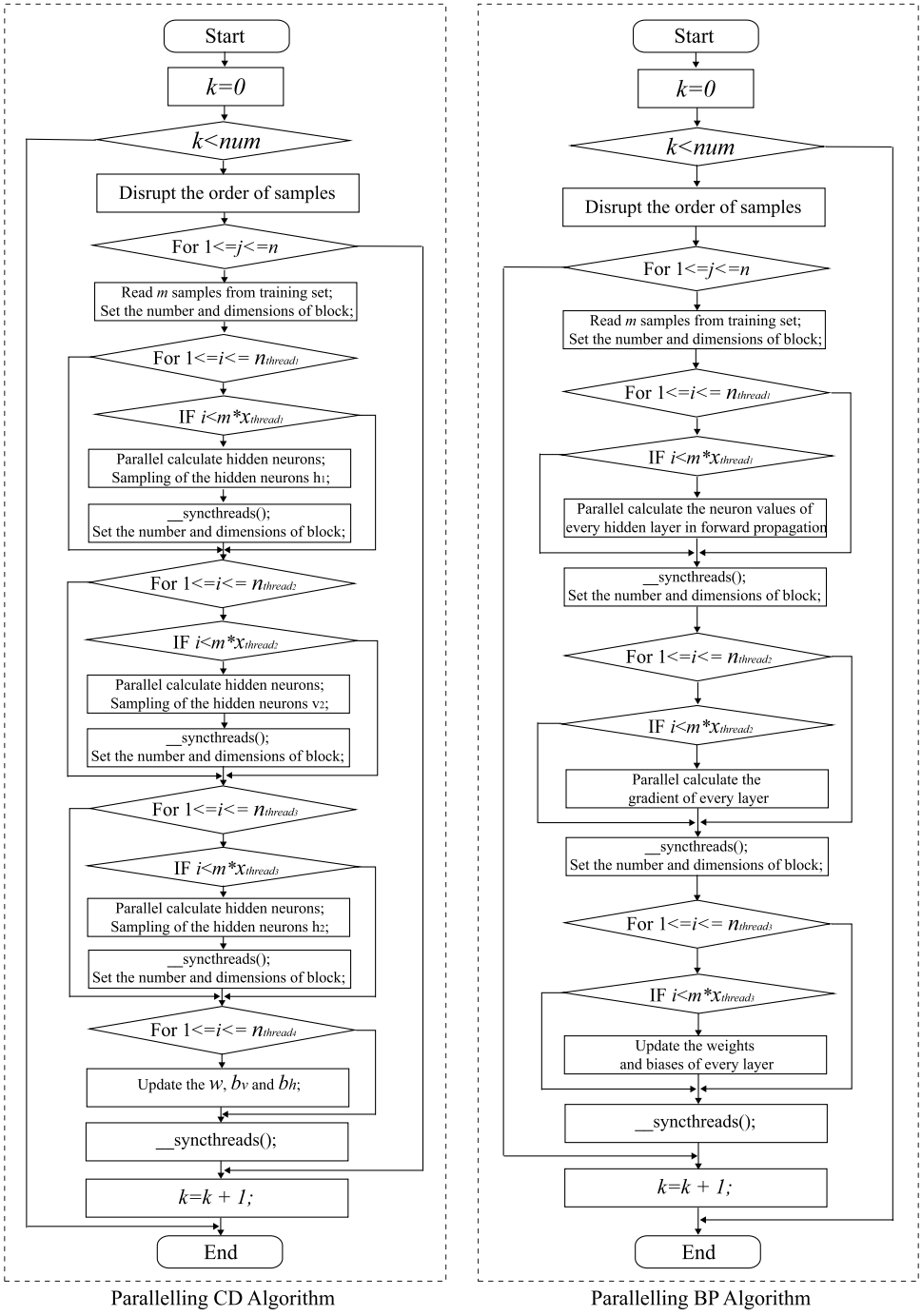

Fig. 5. The training processes of CD algorithm and BP algorithm in the parallel. Multi-layer RBMs obtain the high-level features of the cement image. Fine-tuning via BP algorithm adjusts the parameters of the neural network to estimate CCS.

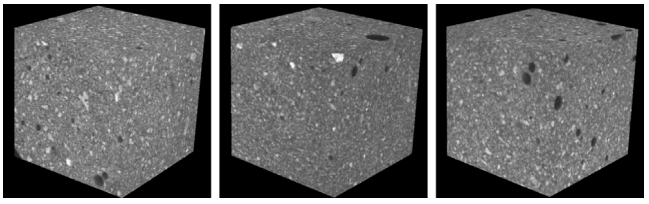

Fig. 6. Three samples of A, B, and C.

samples are tested. In addition, the mean absolute error (MAE) and the absolute error (AE) are the evaluation standard of estimation effect. A small value indicates that the estimation effect is good. Furthermore, the result value is the average of multiple experiments.

Through the trial and error, the DBN consists of three RBMs and one output layer. The number of neurons of hidden layers are 30, 14, and 10. The batch size of each type of cement samples is 10. ALL data set is the three samples and the batch size is 500. The generation of the BP algorithm is set 10000 in all experiments.

Fig. 7 illustrates cement particles in two-dimensional and three-dimensional views, and three-dimensional spatial distribution. In terms of appearance, a distinct difference can be observed between 2-D

Table 2
The features extracted from 3-D cement image.

Features	Description
Mean	The Mean of the gray-level histogram
Variance	The Variance of the gray-level histogram
Skewness	The Skewness of the gray-level histogram
Kurtosis	The Kurtosis of the gray-level histogram
Energy	The Energy of the gray-level histogram
Entropy	The Entropy of the gray-level histogram
μ_{ENT_1}	The Mean(μ) of ENT about GLCM in 13 directions when its distance is 1
μ_{COR_1}	The Mean(μ) of COR about GLCM in 13 directions when its distance is 1
μ_{IDM_1}	The Mean(μ) of IDM about GLCM in 13 directions when its distance is 1
μ_{ENG_1}	The Mean(μ) of ENG about GLCM in 13 directions when its distance is 1
σ_{ENT_1}	The Standard Deviation(σ) of ENT about GLCM in 13 directions when its distance is 1
σ_{COR_1}	The Standard Deviation(σ) of COR about GLCM in 13 directions when its distance is 1
σ_{IDM_1}	The Standard Deviation(σ) of IDM about GLCM in 13 directions when its distance is 1
σ_{ENG_1}	The Standard Deviation(σ) of ENG about GLCM in 13 directions when its distance is 1
μ_{ENT_2}	The Mean(μ) of ENT about GLCM in 13 directions when its distance is 2
μ_{COR_2}	The Mean(μ) of COR about GLCM in 13 directions when its distance is 2
μ_{IDM_2}	The Mean(μ) of IDM about GLCM in 13 directions when its distance is 2
μ_{ENG_2}	The Mean(μ) of ENG about GLCM in 13 directions when its distance is 2
σ_{ENT_2}	The Standard Deviation(σ) of ENT about GLCM in 13 directions when its distance is 2
σ_{COR_2}	The Standard Deviation(σ) of COR about GLCM in 13 directions when its distance is 2
σ_{IDM_2}	The Standard Deviation(σ) of IDM about GLCM in 13 directions when its distance is 2
σ_{ENG_2}	The Standard Deviation(σ) of ENG about GLCM in 13 directions when its distance is 2
μ_{ENT_3}	The Mean(μ) of ENT about GLCM in 13 directions when its distance is 3
μ_{COR_3}	The Mean(μ) of COR about GLCM in 13 directions when its distance is 3
μ_{IDM_3}	The Mean(μ) of IDM about GLCM in 13 directions when its distance is 3
μ_{ENG_3}	The Mean(μ) of ENG about GLCM in 13 directions when its distance is 3
σ_{ENT_3}	The Standard Deviation(σ) of ENT about GLCM in 13 directions when its distance is 3
σ_{COR_3}	The Standard Deviation(σ) of COR about GLCM in 13 directions when its distance is 3
σ_{IDM_3}	The Standard Deviation(σ) of IDM about GLCM in 13 directions when its distance is 3
σ_{ENG_3}	The Standard Deviation(σ) of ENG about GLCM in 13 directions when its distance is 3
μ_{ENT_4}	The Mean(μ) of ENT about GLCM in 13 directions when its distance is 4
μ_{COR_4}	The Mean(μ) of COR about GLCM in 13 directions when its distance is 4
μ_{IDM_4}	The Mean(μ) of IDM about GLCM in 13 directions when its distance is 4
μ_{ENG_4}	The Mean(μ) of ENG about GLCM in 13 directions when its distance is 4
σ_{ENT_4}	The Standard Deviation(σ) of ENT about GLCM in 13 directions when its distance is 4
σ_{COR_4}	The Standard Deviation(σ) of COR about GLCM in 13 directions when its distance is 4
σ_{IDM_4}	The Standard Deviation(σ) of IDM about GLCM in 13 directions when its distance is 4
σ_{ENG_4}	The Standard Deviation(σ) of ENG about GLCM in 13 directions when its distance is 4
μ_{ENT_5}	The Mean(μ) of ENT about GLCM in 13 directions when its distance is 5
μ_{COR_5}	The Mean(μ) of COR about GLCM in 13 directions when its distance is 5
μ_{IDM_5}	The Mean(μ) of IDM about GLCM in 13 directions when its distance is 5
μ_{ENG_5}	The Mean(μ) of ENG about GLCM in 13 directions when its distance is 5
σ_{ENT_5}	The Standard Deviation(σ) of ENT about GLCM in 13 directions when its distance is 5
σ_{COR_5}	The Standard Deviation(σ) of COR about GLCM in 13 directions when its distance is 5
σ_{IDM_5}	The Standard Deviation(σ) of IDM about GLCM in 13 directions when its distance is 5
σ_{ENG_5}	The Standard Deviation(σ) of ENG about GLCM in 13 directions when its distance is 5

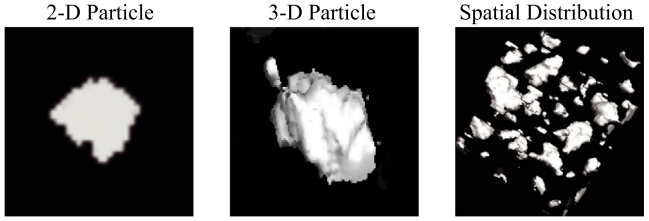

Fig. 7. The appearance of 2-D, and 3-D particle, and spatial distribution.

and 3-D particles. By contrast, 3-D particle has more structural information, which is extremely irregular and complex regardless of spatial structure or spatial distribution. In reality, when the structure is very regular, it is feasible to analyze two-dimensional data for simplifying the calculation. However, when the structure is irregular, the results of 2-D analysis can be quite different from those in three-dimensional case (De Marchis and Napoli, 2012). For cement, the strength is greatly affected by its structures (Wang et al., 2014a), and its structural changes in hydration are also in a three-dimensional way. Therefore, instead of two-dimensional images, it is appropriate to estimate CCS using three-dimensional image.

Since the three-dimensional microstructure images can realistically reflect microstructure and hydration of the cement, this work creatively proposes estimating cement compressive strength using 3-D microstructure images. To test its validity in strength prediction, it is compared with the 2-D image that used in the previous work (Li et al., 2016). In this experiment, both data contain 675 training samples and 135 test samples. Fig. 8 compares the mean absolute error of different methods using the 3-D image and using the 2-D image, respectively. The results illustrate that the prediction error using three-dimensional images is smaller than that using the two-dimensional images on the same methods. For all methods, the MAE is reduced by at least 8.49%. Particularly, both the neural network and the multilayer perceptron are reduced by 18.61%. These phenomena indicate that the three-dimensional microstructure images are more beneficial to estimate the performance of the cement than two-dimensional images.

Table 3 compares the MAE based on deep belief network and other methods on ALL data set, including support vector machine, decision table, linear regression, k-nearest neighbor (Yu et al., 2015), random forest, multilayer perceptron, etc. Moreover, all methods in this experiment show their best performance.

Compared with other methods, the estimation ability of the DBN demonstrates favorable performance for cement strength estimation. Among the other methods, the multilayer perception has a better

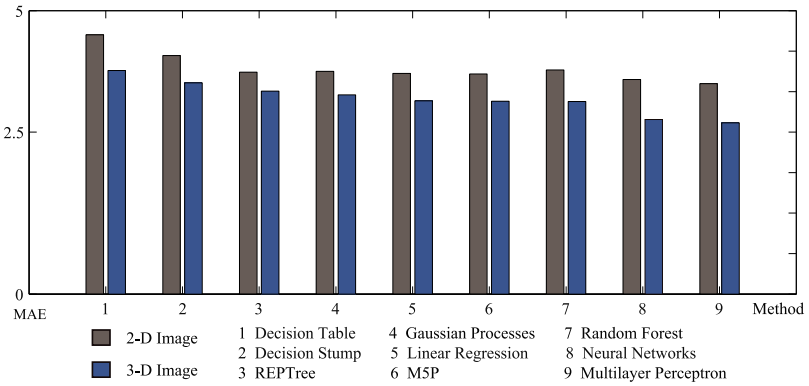

Fig. 8. The comparison of MAE using different methods based on 3-D image and 2-D image.

Table 3
The results of ALL data set in different methods.

Method	MAE	Method	MAE
Support vector machine	3.885957	Decision table	3.884368
Decision stump	3.866422	REP tree	3.84060
Linear regression	3.530127	K-nearest neighbor	3.417005
M5P	3.368629	Random forest	3.27416
Neural networks	3.160597	Multilayer perceptron	2.953329
Deep belief network	**2.809159**		

estimation accuracy, because this method also has multilayers features mapping. This result shows that the nonlinear estimation is better than the linear estimation. Moreover, it also indicates that the relationship between CCS and the images features is nonlinear. The existence of isotropy and homogeneity in cement microstructural image increases the number of local optimum in constructing model. However, the DBN is able to extract and select prominent features from the input data, thus prevent it from being trapped into local optimum (Chen et al., 2015; Uddin et al., 2017). In addition, the fine-tuning of DBN enables minimize error between predictive output and objectives to get optimal value (Fasel and Berry, 2010; Gao et al., 2014). Therefore DBN has better performance than other methods.

In order to prove the estimation effect of this method, the method is compared with two previous researches based on two-dimensional cement image (Li et al., 2017, 2016). Fig. 9(a) presents the predicted results of three researches. It can be found that the results based on two-dimensional image have improved. The predicted effect using convolutional neural network outshines that using a simple neural network. Reassuringly, the experimental result of this work surpasses two previous works, reducing the predicted MAE to 2.8091, which fully demonstrates the effectiveness of this method.

The estimation effect on different data sets is also studied. Fig. 9(b) shows the MAE of different data sets and Table 4 presents the partial results of the AE. As shown in Fig. 9(b), the cement sample A has the best estimation accuracy, while the cement sample B has the worst estimation accuracy. The reason is that the image of sample B has the brightness and the gray-level heterogeneity. The MAE of the three types of the cement sample shows that this method is feasible. Moreover, the results in Table 4 indicate that the data set has a little impact on the estimation accuracy when the generation of the BP algorithm is the same. From the partial results, it can be seen that the AE of most samples are less than 5, and about 40 percent of the samples are within 1, which prove this method has the ability to accurately estimate the strength of the cement.

Since the training process of the deep belief network is time-consuming, its training process is paralleled on the CUDA. In order to verify the parallel effect, this process is run on the CPU and GPU in same machine, respectively. In the experiments, three types of the cement samples are used and the number of each sample is different.

Table 4
The results of three samples in DBN.

Sample	Number	Estimated value (MPa)	Actual value (MPa)	Absolute error
A	1	14.063556	14.025	0.038556
	2	6.714929	6.15	0.564929
	3	14.701393	20.86	6.158607
	4	16.341401	12.8	3.541401
	5	17.539627	14.025	3.514627
	6	16.182942	20.86	4.677058
	7	8.043422	7.6133	0.430122
B	1	18.416169	23.92	5.503831
	2	20.592615	23.92	3.327385
	3	18.759017	23.92	5.160983
	4	10.591213	10.42	0.171213
	5	13.952112	13.81	0.142112
	6	11.78258	10.42	1.36258
	7	21.636857	20.86	0.776857
C	1	10.058224	8.3483	1.709924
	2	15.813328	23.925	8.111672
	3	18.408759	21.0833	2.674541
	4	5.753655	4.9583	0.795355
	5	22.325068	23.925	1.599932
	6	5.796866	4.9583	0.838566
	7	20.30195	14.975	5.32695
ALL	1	8.189257	8.3483	0.159043
	2	18.896861	23.925	5.028139
	3	13.233157	12.9867	0.246457
	4	21.470584	23.925	2.454416
	5	20.730474	13.9867	6.743774
	6	21.60052	23.925	2.32448
	7	21.623956	21.0833	0.540656

The batch size is set to 500 and the BP algorithm is 100 generations. Moreover, the acceleration effect of CUDA is expressed by the speedup ratio.

In order to explore the paralleling effect on the GPU and the influence of sample size on it, large and small samples are used to train and test in this experiment. Moreover, the acceleration effects on two training processes are also studied separately. The time consumption of the RBM and BP algorithm are shown in Fig. 10 and the speedup ratio is compared in Table 5. The results show that this method achieves a 10 times speedup ratio for two processes with same estimation accuracy. Moreover, the speedup ratio in the BP algorithm of large samples is smaller than that of small samples. This reason is that the data transmissions exist between the CPU and GPU when the large samples are trained. These phenomena indicate that this method greatly improves the training efficiency and prove the fact that CUDA programming is necessary when data set is large-scale.

Although the problem of high time consumption can be solved by CUDA environment in parallel, the computation complexity is not really reduced because the calculations in training have not been reduced. Furthermore, its calculation speed depends on the platform, which

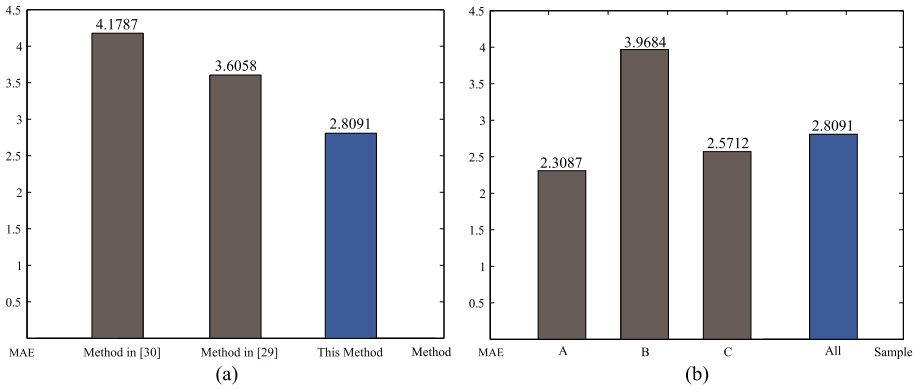

Fig. 9. The MAE of the different research and different data. (a) The comparison of previous research using cement image. (b) The MAE of different data set in DBN.

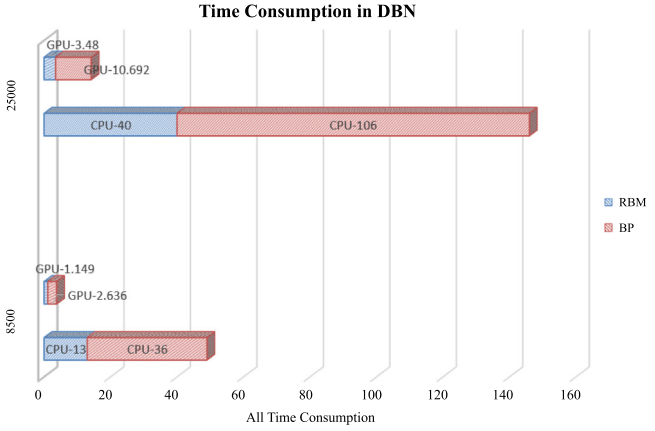

Fig. 10. Time consumption in DBN.

Table 5
Time consumption and speedup ratio in DBN.

Sample number	Method	CPU time (s)	GPU time (s)	Speedup ratio
8 500	RBM	13	1.149	11.31
	BP	36	2.636	13.66
25 000	RBM	40	3.480	11.49
	BP	106	10.692	9.91

is a limitation, but is common for all algorithms based on CUDA environment.

6. Conclusion

This paper proposes an efficient method for estimating cement compressive strength, which estimates the cement compressive strength using three-dimensional microstructure image and deep belief network. In this method, the features representing the phase state and texture information are extracted from the three-dimensional cement images to train estimation model. DBN is adopted to learn and discover patterns between cement images and CCS. Furthermore, the learning processes are parallelized on GPUs to reduce time consumption. Compared with other methods, DBN has multiple layers to learn features and further reduces errors via fine-tuning, thus achieves higher estimation effect. Moreover, its computational time has been greatly reduced for parallelization. Those results imply that the proposed method not only estimates strength nondestructively but also improves the efficiency.

The proposed method reduces the estimation error, but its effect depends on the quality and quantity of the data, as well as network structure and parameters. In addition, it achieves acceleration on CUDA, while the acceleration effect is affected by the platform, which is also the drawback of most acceleration algorithms based on CUDA. Finally, the estimated effect can be further improved since this method adopts the extracted features to estimate CCS, which partially lost the information in the image.

Since extracting features will lose part of the information, the second-degree statistics must be extracted to represent the image features of 3-D microstructure images. Therefore, future work tends to estimate CCS directly based on the 3-D microstructure images and other state-of-the-art methods to avoid information loss in the image feature extraction and to ensure estimation accuracy.

Acknowledgments

This work was supported by National Natural Science Foundation of China under Grant No. 61872419, No. 61573166, No. 61572230, No. 61873324, No. 81671785, No. 61672262. Shandong Provincial Natural Science Foundation, China No. ZR2019MF040, No. ZR2018LF005. Shandong Provincial Key R&D Program, China under Grant No. 2019GGX101041, No. 2018GGX101048, No. 2016ZDJS01A12, No. 2016GGX101001, No. 2017CXZC1206. Taishan Scholar Project of Shandong Province, China under Grant No. tsqn201812077. Project of Shandong Province Higher Educational Science and Technology Program under Grant No. J16LN07.

References

Akkurt, S., Tayfur, G., Can, S., 2004. Fuzzy logic model for the prediction of cement compressive strength. Cem. Concr. Res. 34 (8), 1429–1433.
Bao, W., Yang, B., Li, D., Li, Z., Zhou, Y., Bao, R., 2019. Cmsenn: Computational modification sites with ensemble neural network. Chemometr. Intell. Lab. Syst. 185, 65–72.
Baykasoğlu, A., Dereli, T., Tanış, S., 2004. Prediction of cement strength using soft computing techniques. Cem. Concr. Res. 34 (11), 2083–2090.
Bengio, Y., LeCun, Y., et al., 2007. Scaling learning algorithms towards ai. Large-scale Kernel Mach. 34 (5), 1–41.
Beyeler, M., Oros, N., Dutt, N., Krichmar, J.L., 2015. A gpu-accelerated cortical neural network model for visually guided robot navigation. Neural Netw. 72, 75–87.
Bianchini, M., Scarselli, F., 2014. On the complexity of neural network classifiers: A comparison between shallow and deep architectures. IEEE Trans. Neural Netw. Learn. Syst. 25 (8), 1553–1565.
Bohner, M., Hassan, T.S., Li, T., 2018. Fite–hille–wintner-type oscillation criteria for second-order half-linear dynamic equations with deviating arguments. Indag. Math. (N.S.) 29 (2), 548–560.
Chatzarakis, G.E., Li, T., 2018. Oscillation criteria for delay and advanced differential equations with nonmonotone arguments. Complexity 2018, 1–18.
Chen, C., Jiang, F., Yang, C., Rho, S., Shen, W., Liu, S., Liu, Z., 2018. Hyperspectral classification based on spectralpatial convolutional neural networks. Eng. Appl. Artif. Intell. 68, 165–171.
Chen, C.L.P., Liu, Z., 2018. Broad learning system: An effective and efficient incremental learning system without the need for deep architecture. IEEE Trans. Neural Netw. Learn. Syst. 29 (1), 10–24.
Chen, Y., Liu, J., Hu, Y., Yang, J., Shi, L., Shu, H., Gui, Z., Coatrieux, G., Luo, L., 2017. Discriminative feature representation: an effective postprocessing solution to low dose ct imaging. Phys. Med. Biol. 62 (6), 2103.

Chen, C.L., Zhang, T., Chen, L., Tam, S.C., 2016. I-ching divination evolutionary algorithm and its convergence analysis. IEEE Trans. Cybern. 47 (1), 2–13.
Chen, Y., Zhao, X., Jia, X., 2015. Spectral–spatial classification of hyperspectral data based on deep belief network. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 8 (6), 2381–2392.
Chindaprasirt, P., Jaturapitakkul, C., Sinsiri, T., 2005. Effect of fly ash fineness on compressive strength and pore size of blended cement paste. Cem. Concr. Compos. 27 (4), 425–428.
De Marchis, M., Napoli, E., 2012. Effects of irregular two-dimensional and three-dimensional surface roughness in turbulent channel flows. Int. J. Heat Fluid Flow 36, 7–17.
Doğan, G., Arslan, M.H., Ceylan, M., 2015. Statistical feature extraction based on an ann approach for estimating the compressive strength of concrete. Neural Netw. World 25, 301–318.
Fasel, I., Berry, J., 2010. Deep belief networks for real-time extraction of tongue contours from ultrasound during speech. In: 2010 20th International Conference on Pattern Recognition. IEEE, pp. 1493–1496.
Gao, F., 1997. A new way of predicting cement strength-fuzzy logic. Cem. Concr. Res. 27 (6), 883–888.
Gao, N., Gao, L., Gao, Q., Wang, H., 2014. An intrusion detection model based on deep belief networks. In: 2014 Second International Conference on Advanced Cloud and Big Data. IEEE, pp. 247–252.
Gong, T., Fan, T., Guo, J., Cai, Z., 2017. Gpu-based parallel optimization of immune convolutional neural network and embedded system. Eng. Appl. Artif. Intell. 62, 384–395.
Guo, J., Zhang, N., Wang, L., Yang, B., Zhao, X., Zhou, J., Liu, S., 2016. A fast texture synthesis using gene expression programming. In: 2016 3rd International Conference on Informative and Cybernetics for Computational Social Systems (ICCSS). IEEE, pp. 119–124.
Han, S., Chen, Y., Tang, G., 2017. Fault diagnosis and fault-tolerant tracking control for discrete-time systems with faults and delays in actuator and measurement. J. Franklin Inst. B 354 (12), 4719–4738.
Haralick, R.M., Shanmugam, K., Dinstein, I., 1973. Textural features for image classification. IEEE Trans. Syst. Man Cybern. SMC-3 (6), 610–621.
He, X., Wu, Y., Wu, Y., 2013. Texture feature extraction method combining nonsubsampled contour transformation with gray level co-occurrence matrix. J. Multimedia 8 (6), 675–684.
Hinton, G.E., 2009. Deep belief networks. Scholarpedia 4 (5), 5947.
Hinton, G.E., Salakhutdinov, R.R., 2006. Reducing the dimensionality of data with neural networks. Science 313 (5786), 504–507.
Hinton, G.E., Sejnowski, T.J., 1986. Learning and relearning in boltzmann machines. pp. 282–317.
Hu, C., Min, X., Yun, H., Wang, T., Zhang, S., 2011. Automatic detection of sound knots and loose knots on sugi using gray level co-occurrence matrix parameters. Ann. For. Sci. 68 (6), 1077.
Huang, W., Oh, S., Pedrycz, W., 2017. Fuzzy wavelet polynomial neural networks: analysis and design. IEEE Trans. Fuzzy Syst. 25 (5), 1329–1341.
Jang, H., Plis, S.M., Calhoun, V.D., Lee, J., 2017. Task-specific feature extraction and classification of fmri volumes using a deep neural network initialized with a deep belief network: evaluation using sensorimotor tasks. NeuroImage 145, 314–328.
Khomenko, V., Shyshkov, O., Radyvonenko, O., Bokhan, K., 2016. Accelerating recurrent neural network training using sequence bucketing and multi-gpu data parallelization. In: 2016 IEEE First International Conference on Data Stream Mining Processing (DSMP). pp. 100–103.
Kuremoto, T., Kimura, S., Kobayashi, K., Obayashi, M., 2014. Time series forecasting using a deep belief network with restricted boltzmann machines. Neurocomputing 137, 47–56.
van der Laan, W.J., Jalba, A.C., Roerdink, J.B.T.M., 2011. Accelerating wavelet lifting on graphics hardware using cuda. IEEE Trans. Parallel Distrib. Syst. 22 (1), 132–146.
Larhlimi, A., Mestari, M., 2018. Gpu parallel neural hierarchical multi objective solver for burst routing and wavelength assignment. Eng. Appl. Artif. Intell. 75, 48–63.
Li, X., 2004. Study on Cement Strength Prediction Model Based on Digital Image Processing (Ph.D. thesis). University of Jinan, (in chinese).
Li, Z., Cai, X., Liu, Y., Zhu, B., 2019. A novel gaussian–bernoulli based convolutional deep belief networks for image feature extraction. Neural Process. Lett. 49 (1), 305–319.
Li, M., Wang, L., Yang, B., Zhang, L., Liu, Y., 2017. Estimating cement compressive strength from microstructure images using convolutional neural network. In: 2017 IEEE Symposium Series on Computational Intelligence (SSCI). pp. 1–7.
Li, M., Yang, B., Wang, L., Liu, Y., Zhao, X., Zhou, J., Zhang, L., 2016. The prediction of cement compressive strength based on gray level images and neural network. In: 2016 3rd International Conference on Informative and Cybernetics for Computational Social Systems (ICCSS). IEEE, pp. 103–108.
Lopes, N., Ribeiro, B., 2014. Towards adaptive learning with improved convergence of deep belief networks on graphics processing units. Pattern Recognit. 47 (1), 114–127.
Luo, C., Wu, D., Wu, D., 2016. A deep learning approach for credit scoring using credit default swaps. Eng. Appl. Artif. Intell. 65, 465–470.
Malegori, C., Franzetti, L., Guidetti, R., Casiraghi, E., Rossi, R., 2016. Glcm, an image analysis technique for early detection of biofilm. J. Food Eng. 185, 48–55.
Mohamed, A., Sainath, T.N., Dahl, G., Ramabhadran, B., Hinton, G.E., Picheny, M.A., 2011. Deep belief networks using discriminative features for phone recognition. In: 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 5060–5063.
Murat, H., Zhou, J., Yan, C., Li, L.I., Chen, J., Yanting, H.U., 2012. Feature extraction and analysis on ct image of xinjiang local liver hydatid by using gray-scale histograms. Sci. Technol. Rev. 30 (6), 66–70.
Ni, H., Wang, J., 2000. Prediction of compressive strength of concrete by neural networks. Cem. Concr. Res. 30 (8), 1245–1250.
Peddabachigari, S., Abraham, A., Grosan, C., Thomas, J., 2007. Modeling intrusion detection system using hybrid intelligent systems. J. Netw. Comput. Appl. 30 (1), 114–132.
Ren, Y., Wu, Y., 2014. Convolutional deep belief networks for feature extraction of eeg signal. In: 2014 International Joint Conference on Neural Networks (IJCNN). pp. 2850–2853.
Rios, E., Ochi, L.S., Boeres, C., Coelho, V.N., Coelho, I.M., Farias, R., 2018. Exploring parallel multi-gpu local search strategies in a metaheuristic framework. J. Parallel Distrib. Comput. 111, 39–55.
Rovenskaya, O., Croce, G., 2015. Numerical investigation of the effect of boundary conditions for a highly rarefied gas flow using the gpu accelerated boltzmann solver. Comput. & Fluids 110, 77–87.
Sangdani, M., Tavakolpour-Saleh, A., Lotfavar, A., 2018. Genetic algorithm-based optimal computed torque control of a vision-based tracker robot: Simulation and experiment. Eng. Appl. Artif. Intell. 67, 24–38.
Sarikaya, R., Hinton, G.E., Deoras, A., 2014. Application of deep belief networks for natural language understanding. IEEE/ACM Trans. Audio Speech Lang. Proc. 22 (4), 778–784.
Sebastiá, M., naki Fernández Olmo, I., Irabien, A., 2003. Neural network prediction of unconfined compressive strength of coal fly ashement mixtures. Cem. Concr. Res. 33 (8), 1137–1146.
Seide, F., Li, G., Yu, D., 2011. Conversational speech transcription using context-dependent deep neural networks. In: Twelfth Annual Conference of the International Speech Communication Association. pp. 437–440.
Shang, H., Jiang, Z., Xu, R., Wang, D., Wu, P., Chen, Y., 2019. The dynamic mechanism of a novel stochastic neural firing pattern observed in a real biological system. Cogn. Syst. Res. 53, 123–136.
Sheri, A.M., Rafique, A., Pedrycz, W., Jeon, M., 2015. Contrastive divergence for memristor-based restricted boltzmann machine. Eng. Appl. Artif. Intell. 37, 336–342.
Shi, D., Xu, L., Han, L., 2007. Image retrieval using both color and texture features. J. China Univ. Posts Telecommun. 14, 94–99.
de Siqueira Tango, C.E., 1998. An extrapolation method for compressive strength prediction of hydraulic cement products. Cem. Concr. Res. 28 (7), 969–983.
Son, Y., Lee, S., Kim, H., Song, E., Huh, H., Czosnyka, M., Kim, D., 2018. Automated artifact elimination of physiological signals using a deep belief network: An application for continuously measured arterial blood pressure waveforms. Inform. Sci. 456, 145–158.
Tao, W., Leu, M.C., Yin, Z., 2018. American sign language alphabet recognition using convolutional neural networks with multiview augmentation and inference fusion. Eng. Appl. Artif. Intell. 76, 202–213.
Tsivilis, S., Parissakis, G., 1995. A mathematical model for the prediction of cement strength. Cem. Concr. Res. 25 (1), 9–14.
Uddin, M.Z., Hassan, M.M., Almogren, A., Alamri, A., Alrubaian, M., Fortino, G., 2017. Facial expression recognition utilizing local direction-based robust features and deep belief network. IEEE Access 5, 4525–4536.
Wang, K., Kong, S., 2011. Identification method of waste based on gray level co-occurrence matrix and neural network. In: 2011 International Conference on Materials for Renewable Energy & Environment, Vol. 1. IEEE, pp. 929–931.
Wang, L., Orchard, J., 2018. Investigating the evolution of a neuroplasticity network for learning. IEEE Trans. Syst. Man Cybernet.: Syst. 1–13.
Wang, L., Yang, B., Abraham, A., 2016a. Distilling middle-age cement hydration kinetics from observed data using phased hybrid evolution. Soft Comput. 20 (9), 3637–3656.
Wang, L., Yang, B., Abraham, A., Qi, L., Zhao, X., Chen, Z., 2014a. Construction of dynamic three-dimensional microstructure for the hydration of cement using 3d image registration. Pattern Anal. Appl. 17 (3), 655–665.
Wang, L., Yang, B., Chen, Y., Chen, Z., Sun, H., 2014b. Accelerating fcm neural network classifier using graphics processing units with cuda. Appl. Intell. 40 (1), 143–153.
Wang, L., Yang, B., Orchard, J., 2016b. Particle swarm optimization using dynamic tournament topology. Appl. Soft Comput. 48, 584–596.
Wang, L., Yang, B., Wang, S., Liang, Z., 2015. Building image feature kinetics for cement hydration using gene expression programming with similarity weight tournament selection. IEEE Trans. Evol. Comput. 19 (5), 679–693.
Wang, L., Yang, B., Zhao, X., Chen, Y., Chang, J., 2010. Reverse extraction of early-age hydration kinetic equation from observed data of portland cement. Sci. China Technol. Sci. 53 (6), 1540–1553.
Wang, L., Zhu, X., Yang, B., Guo, J., Liu, S., Li, M., Zhu, J., Abraham, A., 2018. Accelerating nearest neighbor partitioning neural network classifier based on cuda. Eng. Appl. Artif. Intell. 68, 53–62.

Xavier, S., Francisco, M., Victor, U., 2010. Parallel training of a back-propagation neural network using CUDA. In: 2010 Ninth International Conference on Machine Learning and Applications. IEEE, pp. 307–312.

Xie, J., Li, T., Yang, Y., Jin, W., 2014. Learning features from high speed train vibration signals with deep belief networks. In: 2014 International Joint Conference on Neural Networks (IJCNN). pp. 2205–2210.

Yeh, I.-C., 1998. Modeling of strength of high-performance concrete using artificial neural networks. Cem. Concr. Res. 28 (12), 1797–1808.

Yin, J., Zhao, W., 2016. Fault diagnosis network design for vehicle on-board equipments of high-speed railway: A deep learning approach. Eng. Appl. Artif. Intell. 56, 250–259.

Yu, Z., Liu, Y., Yu, X., Pu, K.Q., 2015. Scalable distributed processing of k nearest neighbor queries over moving objects. IEEE Trans. Knowl. Data Eng. 27 (5), 1383–1396.

Zhang, T., Chen, C.L.P., Chen, L., Xu, X., Hu, B., 2018. Design of highly nonlinear substitution boxes based on i-ching operators. IEEE Trans. Cybern. 48 (12), 3349–3358.

Zhang, Y.M., Napier-Munn, T.J., 1995. Effects of particle size distribution, surface area and chemical composition on portland cement strength. Powder Technol. 83 (3), 245–252.

Zhang, T., Su, G., Qing, C., Xu, X., Cai, B., Xing, X., 2019a. Hierarchical lifelong learning by sharing representations and integrating hypothesis. IEEE Trans. Syst. Man Cybernet.: Syst. 1–11.

Zhang, T., Wang, X., Xu, X., Chen, C.L.P., 2019b. Gcb-net: Graph convolutional broad network and its application in emotion recognition. IEEE Trans. Affect. Comput. 1.

Zhang, Q., Yang, B., Wang, L., Zhu, F., 2012. Predicting cement compressive strength using double-layer multi-expression programming. In: 2012 Fourth International Conference on Computational and Information Sciences. pp. 94–97.

Zhou, J., Chen, L., Chen, C.P., Zhang, Y., Li, H.-X., 2016. Fuzzy clustering with the entropy of attribute weights. Neurocomputing 198, 125–134.