Enhancing images of shale formations by a hybrid stochastic and deep learning algorithm

Serveh Kamrava, Pejman Tahmasebi, Muhammad Sahimi

Article history:
Received 19 December 2018
Received in revised form 1 July 2019
Accepted 7 July 2019
Available online 15 July 2019

Keywords:
Deep learning
Stochastic modeling
Shale formation
Imaging

Abstract

Accounting for the morphology of shale formations, which represent highly heterogeneous porous media, is a difficult problem. Although two- or three-dimensional images of such formations may be obtained and analyzed, they either do not capture the nanoscale features of the porous media, or they are too small to be an accurate representative of the media, or both. Increasing the resolution of such images is also costly. While high-resolution images may be used to train a deep-learning network in order to increase the quality of low-resolution images, an important obstacle is the lack of a large number of images for the training, as the accuracy of the network’s predictions depends on the extent of the training data. Generating a large number of high-resolution images by experimental means is, however, very time consuming and costly, hence limiting the application of deep-learning algorithms to such an important class of problems. To address the issue we propose a novel hybrid algorithm by which a stochastic reconstruction method is used to generate a large number of plausible images of a shale formation, using very few input images at very low cost, and then train a deep-learning convolutional network by the stochastic realizations. We refer to the method as hybrid stochastic deep-learning (HSDL) algorithm. The results indicate promising improvement in the quality of the images, the accuracy of which is confirmed by visual, as well as quantitative comparison between several of their statistical properties. The results are also compared with those obtained by the regular deep learning algorithm without using an enriched and large dataset for training, as well as with those generated by bicubic interpolation.

© 2019 Elsevier Ltd. All rights reserved.

1. Introduction

Natural porous media and materials, as well as many synthetic ones, are heterogeneous. In particular, large-scale (LS) porous media are highly disordered over many distinct length scales, from pore to core to much larger scales. Thus, robust characterization of the morphology of porous media, i.e., the spatial distributions of their porosity and the connectivity of their permeable zones, has been a long-standing problem of great interest that has been studied for a long time (Blunt, 2017; Sahimi, 2011). Accurate characterization of the morphology of porous media will not only shed light on the structure of their complex pore space, but will also lead to accurate estimates of their effective flow, transport, reaction, and elastic properties.

With the considerable advancements in instrumentation and measurement techniques, characterization of porous media and materials has made significant progress over the past decade or so. In particular, progress in non-destructive two- and three-dimensional (3D) imaging has made it possible to extract more information on the complexity and heterogeneity of various types of porous media (Brandon & Kaplan, 2008; Herman, 2009; Kak & Slaney, 2001). X-ray imaging (Baruchel et al., 2008; Kinney & Nichols, 1992) eliminates the need for destructive sectioning, and provides high-resolution 3D images for various types of porous media. High-resolution focus ion-beam scanning electron microscopy (FIB-SEM) has also become an essential part of characterization of the microstructure of porous materials, as it reveals important information regarding their morphological properties (Javapour, Fisher, & Unsworth, 2009; Park, Park, & Kang, 2003; Tahmasebi, 2018; Tahmasebi, Javapour, & Sahimi, 2015a, 2015b, 2017).

High-resolution 3D images are not, however, always accessible and, moreover, the scanning process is typically time consuming. It may also not be economical, because one typically requires at least several images, which is costly to obtain. Furthermore, high resolution FIB-SEM can only be used with small-scale samples, which may not be able to capture the essentials of a representative pore-size distribution and other properties of a porous...
sample. Scanning electron microscopy (SEM) provides accurate 2D images of porous media, as it is endowed with the flexibility of field-of-view and resolution. Among other methods for obtaining high-resolution 2D or 3D images, computed tomography (CT), a valuable technique that provides accurate representation of the internal structure of porous material, is increasingly being used (Agar & Geiger, 2015; Jiang, van Dijke, Sorbie, & Couples, 2013). Thus, use of 2D and 3D images for modeling of various types of porous media has been increasing (Tahmasebi, 2018; Tahmasebi et al., 2017; Wang, Arns, Rahman, & Arns, 2018; Wang, Teng, He, Feng, & Zhang, 2018; Wang, Yuan, Rahman, & Arns, 2018).

Shales constitute one of the most complex types of porous media with a multiscale morphology. Due to their ubiquity, they have attracted much attention as the main unconventional source of fossil energy. In addition to their significance as a new source of energy, the complexity of shales’ pore space has also made them the target of a wide range of research, as well as the motivation for the development of new methods of characterizing their morphology, giving rise to a highly active current research field. For example, depending on their location, the morphology of shales can vary widely. They typically present a wide and skewed pore-size distribution, with the sizes of the pores varying from nano- to micro- and mesoscales, which implies that their accurate measurement is difficult. Shales also contain fractures, either natural or hydraulically induced, and the nanopores affect hydrocarbon storage and fluid flow to the micropores and the fractures. At the same time, accurate estimation and/or measurements of shales’ properties is essential to evaluating their gas-storage capacity, and flow and production optimization. High-resolution images of shales contain much details of their morphology, due to their high density of the pixels (voxels in 3D). Therefore, it is important to have high-resolution 2D or 3D images of shales, in order to be able to characterize their pore space and extract important information about the porosity, permeability, mineralogy, total organic carbon (TOC), and other properties. This is more easily said than done, however. One way of addressing this problem is through image enhancement. But, in order for the enhancement to be accurate, one needs an effective approach. In this paper we propose that deep learning can be a powerful tool for this purpose.

In recent years artificial intelligence and deep-learning algorithms have found a wide array of applications. Among such applications, enhancing the quality of images by learning algorithms has had much success. A key reason for the success is the existence of an automatic feature extraction, which is done by training the algorithms with raw data. Deep-learning methods are also considered as algorithms by which nonlinear modules are used in order to form multiple levels of representation. The levels begin from the input data and slowly reach more abstract features. They make the learning process easier for a large number of complex functions, which were previously difficult for the earlier artificial intelligence methods to understand. More importantly, and related to what we study in this paper, is the fact that the power of deep-learning methods has been demonstrated for applications in which one is faced with analyzing large data sets (LeCun, Bengio, & Hinton, 2015). In such situations, such as, for example, when one has multiple arrays of data and complex images, convolutional neural networks (CNNs) are the preferred choice. The main characteristics that make the CNNs distinct are their use of shared weights and biases, pooling, local connections and existence of many layers in the NN (see also below). A convolutional layer identifies local continuity of features from the previous layers and combines them together. The reason for using such architecture is the assumption that some of the features are repeated throughout the image, and that their local statistics are the same, meaning that the same patterns share the same weights and biases (LeCun et al., 2015). In other words, the system that one analyzes is spatially stationary. It is this feature that makes deep learning an attractive method for analyzing problems on porous media.

Use of deep-learning methods can, however, be hampered by lack of large data sets for network training. One way to address the problem is through enhancement of the images that are available for a given porous medium, in order to expand the data set. The enhancement enables one to better estimate the physical properties of a porous medium, such as its permeability, the key property for characterizing fluid flow in its pore space. Therefore, using more robust and accurate image enhancing is crucial. Enhancing images of porous media has been pursued in a few studies in the past that were, however, based on either statistical methods (Gerke, Karsaninia, & Mallants, 2015; Julleh, Rahman, Mccann, Abdullah, & Yeasmin, 2011; Tahmasebi, 2018; Tahmasebi et al., 2015a; Wang, Arns, et al., 2018), or a poor training database (Wang, Arns, et al., 2018).

In this paper, we address the problem of lack of adequate amount of data by proposing a hybrid method. In the proposed approach a stochastic reconstruction method is used to generate numerous images of plausible realizations of a porous medium as the input data based on a few initial images. It then uses various filters in order to generate further variations in the stochastically-generated images. The diverse dataset is then used to train the deep-learning network by linking the high- and low-resolution segments of the images. Practically speaking, preparing a single high-resolution image of shale samples takes a considerable amount of time – at least a few days – which includes cutting, cleaning and drying the sample, assembling the core and core holder, and imaging and processing the raw data. Aside from being very costly, the process is also very time consuming. Furthermore, the imaging tools typically have their limitations in terms of capturing the fine-scale information that are vital for evaluating the physical properties. In this paper we combine a physics-based approach for reconstructing models of porous media with deep learning to develop an application of the latter in rock physics.

The rest of this paper is organized as follows. In the next section we discuss briefly the background on deep learning and stochastic modeling. In Section 3 the hybrid method that consists of the stochastic modeling of porous media and deep-learning methods is described and utilized to generate various images. Next, the high- and low-resolution images are compared based on several statistics and morphological characteristics of the porous media. The paper is summarized in the last section.

2. Methodology

Enhancing low-resolution and noisy images has been addressed in the literature (Glasner, Bagon, & Irani, 2009; Kim, Lee, & Lee, 2015; Park et al., 2003). The goal has been increasing the image quality by minimizing the mean-square errors (MSE) between the generated high-resolution image and the original low-quality version (Ledig et al., 2016). Given the available data, the problem can be addressed by using either a single image or several of them in order to produce enhanced images. When only a single image is used, the result is not usually very accurate because it does not contain the high-frequency contents, which is clearly due to the fact that a low-quality image does not contain the necessary information for enriching the image further. In addition, such methods produce images with limited improvement in their quality relative to the original low-resolution image. Moreover, the problem as described is ill-posed in the sense that a single low-resolution image can generate many high-resolution ones. The problem is more complex when the high-quality image...
must be generated based on a low-resolution image that contains little, if any, of the high-frequency features.

An important problem is upscaling of models of porous media in which a high-resolution model involving several millions of blocks in the computational grid that represents the porous media is coarsened to produce another model that requires less intensive computations, but is just as accurate as the original high-resolution model (Ebrahimi & Sahimi, 2004; Mehrabi & Sahimi, 1997; Rasaei & Sahimi, 2008; Rezapour, Ortega, & Sahimi, 2019 and references therein). Clearly, a high-resolution image must be represented by a fine-resolution computational grid that requires, however, a considerable amount of computation time. An accurate upscaling method (see, for example Rasaei & Sahimi, 2009a, 2009b; Rezapour et al., 2019) preserves the important information in the high-resolution computational grid and coarsens the rest. On the other hand, at larger length scales the heterogeneity of the pore space is much more severe, but information about it is missing in the low-resolution image.

2.1. Deep learning

Although machine-learning methods have had many successful applications to various problems, their application to modeling of porous media has not been as fruitful, particularly in cases in which the data are represented by images, due to the limited ability of the methods for processing natural data. For example, pixel values in an image must first be transformed into a feature vector that can be detected by the method. Then, a new class of machine-learning algorithms – the deep-learning methods – was developed in early 2006 that are trained by multiple levels of representation in order to model complex correlations between the various input data. Raw data can then be used as the input to accomplish such important goals as detection or classification of certain features in data sets. Deep feed-forward networks, which are also known as the CNNs, represent typical deep-learning methods (Dahl, Sainath, & Hinton, 2013; Deng & Dong, 2014; Kim, 2016; LeCun et al., 2015; Schmidhuber, 2014; Wu, Jiang, Wang, Pu, & Xue, 2014).

In the context of improving the quality of images, deep-learning algorithms learn the mapping between low- and high-resolution images that differ only in the high-frequency details through the training process. Indeed, the performance of every method in this field depends on how the network is trained, which is strongly controlled by the size and diversity of the data set provided for the training (Dong, Loy, He, & Tang, 2016). Therefore, one prominent issue in using such a method is its requirement for having many high-resolution images for training the network (Dong et al., 2016; Johnson, Alahi, & Fei-Fei, 2016; Kim et al., 2015; Mao, Shen, & Yang, 2016). On the other hand, having large sets of high-resolution images for complex porous media is not feasible, because obtaining them is time consuming and costly. To alleviate the issue, we propose a method to use a limited number of high-resolution images, as few as one image, to generate a diverse large dataset for the training phase of the deep learning. Previous studies that utilized CT images with deep-learning methods had either focused on optimizing the parameters used in the network, or on evaluating the quality of the output image (Hagita, Higuchi, & Jinmai, 2018; Wang, Teng, et al., 2018). To our knowledge, however, deep-learning algorithms using a limited number of actual images have not been proposed or implemented.

2.2. Stochastic modeling

The main purpose of using a stochastic modeling is increasing the number of training images and diversifying the patterns. For this aim, the cross correlation-based simulation (CCSIM) algorithm is used, which has been shown to successfully model various complex 2D and 3D porous media based on their images (Tahmasebi & Sahimi, 2012, 2013, 2016a, 2016b). The CCSIM represents the digital image DI or the porous medium to be modeled, by a computational grid G, partitioned into overlapped blocks of sizes $T_x \times T_y$, where G and DI have the same sizes. The neighboring blocks share overlap regions OL with sizes $\ell_x \times \ell_y$; see Fig. 1. One then begins from a corner block of G (or any other block) and visits each grid block along a one-dimensional raster path. For each grid block a pattern of heterogeneity from DI is selected randomly and inserted in the visiting block. We refer to the inserted pattern as the data event $D_T$, with the word “event” implying that the inserted pattern of the heterogeneity in the block may change again (see below). Then, the next pattern is selected based on the similarity between the neighborhood points and the DI, meaning that, instead of considering all the previously constructed blocks (by filling them with patterns of heterogeneity from DI), only those in the neighborhood of the current blocks are used for the calculations. Next, the similarity between, or closeness to, the neighboring blocks and the DI is quantified based on the cross-correlation function that represents a convolution between DI and $D_f(x, y)$:

$$
ψ(i, j; x, y) = \sum_{x=0}^{\ell_x-1} \sum_{y=0}^{\ell_y-1} D(x + i, y + j)D_f(x, y),
$$

(1)

with $i \in [0, T_x + \ell_x - 1]$ and $j \in [0, T_y + \ell_y - 1]$.

Thus, an overlap region of size $\ell_x \times \ell_y$ between two neighboring blocks and a data event $D_f$ are used to match the patterns in the DI. The overlap region contains a set of pixels or voxels that we pick from the previously constructed blocks and utilize them in Eq. (1) for identifying the next pattern of heterogeneity. For Euclidean distance (difference) between the constructed block and the data to be minimum, $ψ(i, j; x, y)$ must be maximum or, in practice, exceed a preset threshold. After calculating $ψ(i, j; x, y)$ and selecting those patterns for which $ψ(i, j; x, y)$ exceeds the preset threshold, one of the acceptable ones is selected at random and inserted in the block currently being visited in G. The process is repeated until all the blocks of the grid G have been reconstructed. As a rule of thumb, the neighboring regions might have an overlap of size of about 1/5−1/6 of the size of the blocks. Large grid blocks increase the computations, as computing $ψ(i, j; x, y)$ would require considering many points, whereas small regions may cause discontinuity in the patterns.

To demonstrate the accuracy of the CCSIM method, we generated three realizations of an image. The results are shown in Fig. 2, indicating that the proposed method can utilize the given 2D image and produce the realizations that are not only different from each other, but also from the input image, while their structures are statistically similar. Thus, using this strategy, one can produce a large number of realizations for training as the input.

Fig. 2 also indicates the advantage of using the CCSIM algorithm, namely, the ease of extending it to 3D images. The computations associated with the CCSIM method may also be carried out in the frequency domain, which would result in significant acceleration of the calculations (Tahmasebi, Sahimi, & Carr, 2014). Thus, the algorithm may be used straightforwardly for generating thousands of images in a reasonable time.
3. Hybrid stochastic deep-learning algorithm

Schematic representation of the proposed hybrid stochastic deep-learning (HSDL) algorithm is presented in Fig. 3. As already pointed out, large data sets are essential to the accuracy of deep learning (Angermueller, Pärnamaa, Parts, & Stegle, 2016), but it is not always possible to have such sets. Thus, to remedy this, we used a limited number of images – 30 2D images with a size of $500 \times 500$ – with the stochastic CCSIM algorithm in order to generate 2000 plausible realizations of the same, which are used as the training images in the deep-learning algorithm. The size of 500 of the output realizations was $1000 \times 1000$, with the rest having the same size as the original images. Although the success of any learning algorithm depends heavily on the number and diversity of the images that are used for its training, using similar images may in fact cause overfitting. To diversify, the 2000 training images were transformed using various filters, such as the Gaussian noise, scale, crop, rotation, and flip, to increase the diversity between the training datasets. The details are as follows.

The images were randomly divided into 20 sets. Gaussian noise with three different variances was applied to the images in three sets. Two sets with images of size $500 \times 500$ were enlarged and then cropped back to the original size of the DI, $500 \times 500$. Similarly, images from two other sets that contained larger stochastic images of size $1000 \times 1000$, generated by the CCSIM method, were rescaled to smaller sizes and cropped to have the same size as rest of the DI. We used three scale factors for enlarging and shrinking the training images, because we assumed that we have no information about the scale discrepancy between the input and training images. As such, it is reasonable to include a wide range of scales in the training data. Furthermore, such a diverse scaling helps the network to better learn and capture the multiscale structure of the images.

Flipping was used with the images in three sets. They were flipped left–right, up–down and left–right, and up–down. Then, cropping was applied to the resulting images in the three sets. As mentioned earlier, a portion of the stochastic training images was generated with larger dimensions in order to have the same quality when some of the filters, such as rotation or resizing to smaller sizes, were applied to them. Therefore, rotation and resizing were applied to such images as well. Fig. 4 presents the input images and examples of the results after applying the filters. As can be seen, the filters changed the initial input. Such variations in the input training images help a deep-learning algorithm to perform better in predicting the features of a high-resolution image, because the low-resolution input image misses many features when it is up-scaled to the target size. Many of such patterns might, however, be repeated in the training phase and, thus, a trained network can retrieve the missing features using the high-quality images.

The deep-learning algorithm uses the high-quality training data and extracts their differences with the up-scaled images of the same set, using a residual learning strategy. In fact, the training images all have high resolution, whereas the image at hand that requires enhancement is a low-resolution one, i.e., one with a smaller size. Therefore, the target image is enlarged, i.e., up-scaled to the size of the training images by using an interpolation method with bicubic functions. Then, the network learns iteratively how to estimate the residuals. After the training is complete, i.e., after the network learns how to estimate the residuals, the high-quality image is reconstructed by adding the original enlarged image to the estimated residuals. Therefore, the up-scaled images are used as the input, while the estimated residuals represent the output.

The deep-learning neural network that we used had 22 individual layers, whose architecture is shown in Fig. 3. It represents a convolutional deep-learning network that learns mapping of high-resolution training images onto the low-resolution input image. The utilized images are similar in their content, but differ in their details. To reduce the computational time, 64 random patches of size $41 \times 41$ from the training data were selected from each training image and used in the neural network, rather than working with the full-size images. One may argue that...
such patches can jeopardize capturing large-scale structures in the images. Using the aforementioned filters, however, and in particular the scaling operators, the large-scale structures were still accounted for. Then, the constructed small patches were fed to the neural network over several iterations. If the computational power can be afforded, one can, of course, use the entire images in the computations.

The low-resolution image represents the first layer. The next layer is the convolution layer, which in this study contained 64 filters of size 3 × 3, hence assigning one filter to each patch. The performance of the deep-learning networks improves with increasing the number of the filters, but the training time also increases as well. Choosing a smaller filter size is usually preferred in the sense that it reduces the computational time, but it may also result in missing the large-scale structures of the image. On the other hand, larger filter sizes complicate the training process (Wang, Teng, et al., 2018). Thus, the optimal filter size should be decided a priori, or selected by trial and error. Zero padding was also used in our study in order to generate the same input layer. Every convolution layer, except the last one, had 64 filters of size 3 × 3. The remaining 19 layers contained similar convolutional rectified linear unit (ReLU) elements. The initial weights were assigned randomly, but were optimized during the learning. For the hidden layers we used the ReLU elements, which represent the simplest form of nonlinear activation function, instead of the usual sigmoid. The layer applied the following equation – the rectifier – to the input values without changing its depth information:

\[ f(x) = \log[1 + \exp(x)] , \]

which is called the Softplus or SmoothReLU function. In effect, deep learning takes the extracted features at each pixel x and yields f(x). The ability of the ReLU for speeding up the computations with large training networks and making it much faster than the common activation function has already been demonstrated (Nair & Hinton, 2010). The last layer of the network has a single filter of size 3 × 3 × 50, followed by a regression layer that calculates the MSE σ² between the estimated residual image and the actual one available in the training data:

\[ \sigma^2 = \frac{\sum_{i,j,k} (y_{\text{Resid}}(i,j,k) - y_{\text{HDSL}}(i,j,k))^2}{\sum_{i,j,k} y(i,j,k)} \]
the slower does the iteration travel along the downward slope toward the true minimized state. While using low initial values of the learning rate may seem appealing in terms of making sure that no local minima is taken as the true one, the optimization computation will also take a long time to converge, especially if it is trapped on a plateau region.

Table 1 summarizes the values of all the parameters used in the HSDL computations, most of which were estimated by starting with an initial guess for each parameter and iterating and refining the estimates by straightforward computations. We used an initial learning rate of 0.1 and a gradient threshold of 0.01. The learning rate was decreased to $10^{-10}$ in 100 epochs, using the indicated learning rate factor. $L_2$ norm of the gradients was used as the gradient threshold method. The training was carried out on a GTX-1030 graphic card (NVIDIA) and took about 41 GPU hours and 193,000 iterations.

The loss function for the training and validation phases are compared in Fig. 5. They both reduce drastically, reaching their final value after around 20 epochs. Note, however, that due to the dropout layers in the network, the noise in the loss function for both training and validation is omitted. Fig. 5 also indicates that the number of epochs used, 100, is sufficient for training the network.

4. Results and discussion

The HSDL algorithm was used to analyze and model a complex shale formation with irregular pores. To check the accuracy of the
results, they are compared visually and computationally based on rigorous statistical tests, which measure the similarity between the enhanced images and the original one. To better demonstrate the capability of the proposed method, the results are compared with one of the common image-resizing methods, namely, the bicubic interpolation method, as well as with the regular deep-learning algorithms. Next, the accuracy of the generated images is compared in terms of the connectivity and correlation function of the images.

4.1. Comparison of the images’ features

Fig. 6 presents visual comparison of the results, indicating excellent accuracy of the proposed HDSL algorithm. The initial low-resolution image represents a very smooth and opaque view of the pores in the shale sample, whereas the enhanced image generated by the HDSL manifests features as seen in the reference image. Similarly, the image generated by the bicubic interpolation method is a smooth reconstructed image. As expected, the regular deep-learning algorithm does not perform well when only a limited number of images is used. Note, however, that in modeling of porous media one usually has only a few high-resolution images, which are not sufficient for taking advantage of the recent artificial intelligence methods. A better comparison between the results generated by the various methods is provided by the zoom-in portions in the images; these are also shown in Fig. 6. Note that no similar image, in addition to the low-resolution input, was used in the training of the HDSL network.

We also compared the results quantitatively using various statistical measures. The first measure we used was peak signal-to-noise ratio (PSNR) \( \bar{R} \), which was used with the images generated by both the bicubic interpolation and the HDSL algorithm, and compared with that of the reference high-resolution image. \( \bar{R} \) is defined by

\[
\bar{R} = 10 \log_{10} \left( \frac{M_l^2}{\sigma^2} \right)
\]

where \( M_l \) denotes the maximum possible pixel value in an image \( I \). The second measure is the structural similarity index \( \bar{I}_s \) (sometimes referred to as the SSIM) that evaluates the visual impact of the characteristics of an image, such as the luminance \( I \), the contrast \( c \), and the structure \( s \) against the high-resolution image. It is computed by Wang, Bovik, Sheikh, and Simoncelli (2004),

\[
\bar{I}_s(I_1, I_2) = \frac{(2\mu_I \gamma + C_1)(2\sigma_{xy} + C_2)}{(\mu^2_I + \sigma^2_I + C_1)(\sigma^2_{xy} + \sigma^2_y + C_3)}
\]

where \( I_1 \) represents the enhanced image while \( I_2 \) is the high-resolution image. Consider two windows \( x \) and \( y \) that may be two complete images. Then, the three aforementioned quantities are given by

\[
\bar{I}_l(I_1, I_2) = \frac{2\mu_I \mu_y + C_1}{\mu^2_I + \mu^2_y + C_1},
\]

\[
c(I_1, I_2) = \frac{2\sigma_{xy} + C_2}{\sigma^2_x + \sigma^2_y + C_2},
\]

\[
s(I_1, I_2) = \frac{\sigma_x + C_1}{\sigma_x + C_3}.
\]

Here, \( \mu \) and \( \sigma \) represent, respectively, the mean and standard deviations of the pixel values in the indicated windows, \( \sigma_{xy} \) is the covariance of \( x \) and \( y \), \( C_1 \) and \( C_2 \) are two variables to stabilize the above ratios with weak denominator, and \( C_3 = C_2/2 \) is the dynamic range of the pixel values, which is 1, and typically, \( k_1 = 0.01 \) and \( k_2 = 0.03 \). In the limit, \( \alpha = \beta = \gamma = 1 \) one obtains

\[
\bar{I}_l(I_1, I_2) = \frac{(2\mu_I \mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu^2_I + \mu^2_y + C_1)(\sigma^2_{xy} + \sigma^2_y + C_3)}.
\]

Another quantitative comparative measure is the natural image quality evaluator (NIQE) that estimates the perceptual image quality, measuring the distance between a natural scene's statistics of the input image and features of an image in the data set used to train the HSDL network (Mittal, Soundararajan, & Bovik, 2013). Multidimensional Gaussian distributions were used to model the features in our study.

The computed results for the numerical measures are listed in Table 2. Larger values of the PSNR \( \bar{R} \) and SSIM \( \bar{I}_s \) indicate better image quality. But, while higher values of \( \bar{R} \) represent closer similarity between the enhanced image and the reference one, they are based on pixels’ error and do not consider how well, in order to include the contrast, brightness and structure of the image. Lower values of the NIQE represent better perceptual quality, and are interpreted as implying a smaller distance – closer similarity – between the natural scene of the generated image and the reference image.

Table 2 indicates that the highest values of the PSNR and SSIM along with lowest value of the NIQE are produced by the images generated by the HDSL image. Moreover, the quantitative comparison in Table 2 indicates that the regular deep-learning algorithm without enriched training image produces less accurate results than those produced by the HDSL algorithm.

We also evaluated the accuracy of the HDSL algorithm by comparing the frequency distributions of the pixels. Fig. 7 presents

---

**Table 1** Summary of the parameters used in the HDSL algorithm.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of training images</td>
<td>2000</td>
</tr>
<tr>
<td>Minibatch size</td>
<td>64</td>
</tr>
<tr>
<td>Initial learning rate</td>
<td>0.1</td>
</tr>
<tr>
<td>Learning rate factor</td>
<td>0.1</td>
</tr>
<tr>
<td>Maximum epoch</td>
<td>100</td>
</tr>
<tr>
<td>Gradient threshold</td>
<td>0.01</td>
</tr>
<tr>
<td>Momentum coefficient</td>
<td>0.9</td>
</tr>
<tr>
<td>( \beta )</td>
<td>0.5</td>
</tr>
</tbody>
</table>

---

**Table 2** Computed PSNR, SSIM and NIQE for the bicubic, regular deep learning, and HDSL algorithm.

<table>
<thead>
<tr>
<th>Method</th>
<th>PSNR</th>
<th>SSIM</th>
<th>NIQE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bicubic interpolation</td>
<td>25.775</td>
<td>0.7017</td>
<td>6.133</td>
</tr>
<tr>
<td>Regular deep learning</td>
<td>25.546</td>
<td>0.7031</td>
<td>5.821</td>
</tr>
<tr>
<td>HDSL</td>
<td>26.0586</td>
<td>0.7094</td>
<td>5.454</td>
</tr>
</tbody>
</table>
the comparison between the results for the low- and high-resolution images, as well as those generated by the bicubic interpolation and the HSDL algorithm. The results indicate that the frequency distribution of the image produced by the HSDL algorithm is the closest to that of the reference image.

4.2. Comparison of the images’ morphology

Similar to any type of porous media and materials, accurate models of shale formations are crucial to the study of flow of water, oil, and gas in them. Developing such models entails the ability to correctly represent the morphology of the pore space, including its porosity – the volume fraction of the pores – the long-range connectivity of the pores that provide the paths for fluid flow through the formation, and the correlations between them. Table 3 summarizes the results for the porosity, which confirm that the estimate for the porosity of the image generated by the HSDL algorithm is very close to that of the original high-resolution reference image.

Next, we characterized the spatial long-range connectivity of the images using the multiple-point connectivity (MPC) function

![Comparison of images](image_url)

Table 3: Comparison between estimates of the effective porosity of the various images.

<table>
<thead>
<tr>
<th>Image type</th>
<th>Porosity (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reference image</td>
<td>3.78</td>
</tr>
<tr>
<td>Input image</td>
<td>2.33</td>
</tr>
<tr>
<td>Bicubic interpolation</td>
<td>2.44</td>
</tr>
<tr>
<td>HSDL</td>
<td>3.67</td>
</tr>
</tbody>
</table>
Fig. 7. Frequency distributions (histograms) for the low-resolution (LR) input, high-resolution (HR) reference, bicubic interpolation, and HSDL images.

(Krishnan & Journel, 2003). The MPC function is the probability \( p(r; s) \) of having a sequence of \( s \) connected points in the pore space in a given direction \( r \). If an indicator function \( I(u) \) is defined for the pores of a porous formation by

\[
I(u) = \begin{cases} 
1 & \text{for } u \in \text{pore space} \\
0 & \text{otherwise} 
\end{cases}
\]  

(12)

then, \( p(r; s) \) is defined by

\[
p(r; s) = \text{Prob}\{ I(u) = 1, I(u + r) = 1, \ldots, I(u + sr) = 1 \} 
\]

(13)

Thus, we computed \( p(r; s) \) for the reference high-resolution image, those obtained by the HSDL and bicubic methods, and for the original low-resolution image. The results are shown in Fig. 8. Clearly, the results of the original high-resolution image and the enhanced one produced by the HSDL algorithm are practically identical.

The third morphological property that we used to test the accuracy of the method is the autocorrelation function \( g(r) \), defined by

\[
g(r) = \frac{\langle [I(u) - \phi][I(u + r) - \phi] \rangle}{\phi - \phi^2} 
\]

(14)

where \( r = |r| \), \( \phi \) is the porosity of the porous formation, and \( \langle \cdot \rangle \) denotes a spatial average over locations \( u \) and, therefore, \( \phi = \langle I(u) \rangle \). Fig. 9 compares the computed autocorrelation functions for the four images. Once again, the agreement between the autocorrelation functions of the original high-resolution image and the one that was generated by the HSDL algorithm is excellent.

5. Summary

Producing high-quality images of porous media, and in particular shale formations, by conventional experimental methods is a difficult task that requires considerable investment of time and resources. Most of the currently-available images are of low-quality type and require significant enhancement, if they are to be used in practice for modeling of shale formations. One may train a deep-learning network by the available data or images and then utilize it to enhance the resolution of the available low-resolution images. The training faces a major difficulty, however, as it requires a large number of images that are usually unavailable.

To address the problem of enhancing the resolution and accuracy of images of porous media with a limited number of images, we proposed a novel hybrid method. First, a reconstruction method is used to generate a large number of plausible realizations of a shale formation based on a limited number of high-resolution images, which is accomplished at very low computation cost. They are then used to train a deep-learning network. The results were compared with the enhanced images generated by the bicubic interpolation and the reference image, and those produced by the network without the enriched dataset. The comparison confirmed the superior quality of the HSDL-generated images, when compared with those produced by bicubic interpolation and regular deep learning. Therefore, the images produced by the HSDL algorithm will enable one to better estimate the physical properties of complex porous media, including shale formations, and simulate and forecast flow of multiphase fluids in them.

Fig. 8. Comparison of the computed multi-point connectivity function \( p(r) \). Solid black and gray, dashed gray and dotted gray represent, respectively, the results for the reference high-resolution, HSDL, bicubic interpolation, and the original low-resolution images. The results for the original high-resolution image and the enhanced image produced by the HSDL algorithm are practically identical and difficult to separate.


