*Data-Centric Engineering* (2023), 4: e21
doi:10.1017/dce.2023.16

CAMBRIDGE UNIVERSITY PRESS

RESEARCH ARTICLE

# Content-based image retrieval for industrial material images with deep learning and encoded physical properties

Myung Seok Shim[1], Christopher Thiele[2], Jeremy Vila[3], Nishank Saxena[3] and Detlef Hohl[3]

[1]Shell Global Solutions (US), Inc., Houston, TX, USA
[2]ERGO Group AG, Düsseldorf, Germany
[3]Shell International E&P, Inc., Houston, TX, USA
**Corresponding author:** Myung Seok Shim; Email: Myung-Seok.Shim@shell.com

**Received:** 15 December 2022; **Revised:** 03 August 2023; **Accepted:** 03 August 2023

**Keywords:** Digital rock; image retrieval; materials images; Siamese neural network; similarity learning

## Abstract

Industrial materials images are an important application domain for content-based image retrieval. Users need to quickly search databases for images that exhibit similar appearance, properties, and/or features to reduce analysis turnaround time and cost. The images in this study are 2D images of millimeter-scale rock samples acquired at micrometer resolution with light microscopy or extracted from 3D micro-CT scans. Labeled rock images are expensive and time-consuming to acquire and thus are typically only available in the tens of thousands. Training a high-capacity deep learning (DL) model from scratch is therefore not practicable due to data paucity. To overcome this "few-shot learning" challenge, we propose leveraging pretrained common DL models in conjunction with transfer learning. The "similarity" of industrial materials images is subjective and assessed by human experts based on both visual appearance and physical qualities. We have emulated this human-driven assessment process via a physics-informed neural network including metadata and physical measurements in the loss function. We present a novel DL architecture that combines Siamese neural networks with a loss function that integrates classification and regression terms. The networks are trained with both image and metadata similarity (classification), and with metadata prediction (regression). For efficient inference, we use a highly compressed image feature representation, computed offline once, to search the database for images similar to a query image. Numerical experiments demonstrate superior retrieval performance of our new architecture compared with other DL and custom-feature-based approaches.

## Impact Statement

A dramatic reduction in cost of computing and storage has led to a huge increase in volume of images acquired and intensity of image analysis. In this contribution, we focus on "digital rocks", 2D and 3D images scanned from subsurface porous rock samples that are used for oil and gas reservoirs evaluation, in mining, or in planning of subsurface storage sites for CO2 and hydrogen. Finding rock images similar to a newly acquired sample image with the proposed Double Siamese Neural Network can be done in seconds, compared to weeks or months required by experimentation on rock samples. This orders-of-magnitude acceleration equates to millions of dollars in monetary terms for each sample, and in higher values for missed industry opportunities.

## 1. Introduction

Content-based image retrieval (CBIR) is an established application in computer vision (Gudivada and Raghavan, 1995; Devareddi and Srikrishna, 2022; Kulkarni and Manu, 2022). The objective is to retrieve images similar in content to a query image from a large image database. Following the success of deep learning (DL) in other computer vision applications (Voulodimos et al., 2018), DL has almost entirely replaced targeted feature engineering and greedy-based matching approaches (see Alcantarilla et al., 2012 for a particularly sophisticated incarnation) as the dominant technical approach for CBIR. Multiple comprehensive and comparative reviews demonstrate this in a variety of applications (Zagoruyko and Komodakis, 2015; Zheng et al., 2018; Rian et al., 2019; Chen et al., 2022; Simran et al., 2021; Dubey, 2022). The majority of CBIR studies use common photographic images, and new approaches are almost always developed and calibrated in this rich public domain.

Providing efficient CBIR in an industrial image domain allows experts to quickly infer properties about the query image without the need for expensive or time-consuming physical experimentation. Industrial images provide different challenges, and have different requirements and properties. While state-of-the-art DL models on photographic images can be trained on datasets in the millions, industrial image databases are usually smaller because the images are more costly to collect and curate. This lends itself to DL architectures of smaller capacity. Some prior work addresses the use of databases and convolutional neural networks (CNNs) for analyzing images of common industrial materials (e.g. segmentation and classification in Bell et al., 2015; Ahuja et al., 2022), and the medical community is routinely using DL-based CBIR (Pradhan et al., 2021). However, CBIR applications in the industrial and materials domain are scarce.

In this article, we consider a database of "digital rock" images acquired in 2D via light microscopy or projected from 3D microcomputer tomography to 2D (Saxena et al., 2019b). They are of fundamental importance in the Earth science domain and have led to a revolution in the understanding of deep-Earth phenomena, such as fluid transport through porous media (Blunt et al., 2002; Andrä et al., 2013; Al-Marzouqi, 2018; Saxena et al., 2019a, 2019b). Computer simulation and DL-based image analysis have already greatly advanced the field of rock and fluid science (Araya-Polo et al., 2018; Saxena et al., 2021; Ahuja et al., 2022). Here, we focus on the specific CBIR challenges presented by the materials imaging domain with digital rock as a high-value example. We focus on resolving the following critical issues: (a) data paucity, (b) "expert knowledge" subjectivity, and (c) uncommon statistical image properties.

To solve the data paucity issue, we use DL models that are pretrained on common photographic images in conjunction with "few-shot" transfer learning (Wang et al., 2020). A model pretrained on several related tasks is fine-tuned with limited new data which helps the model generalize to unseen samples. Rather than solely relying on expert subjectivity in image similarity labeling, we use the metadata from physical measurements as additional labels for training. We present a novel custom DL architecture that combines Siamese neural networks where two neural networks share the same weights and produce comparable output vectors for similarity learning (Bromley et al., 1993; Wiggers et al., 2019; Chicco, 2021; Ahuja et al., 2022) and the efficient ResNet architecture (He et al., 2016). We train this Double Siamese Neural Network (DSNN) with a loss function that integrates classification and regression terms to train with both image and metadata similarity (Thiele et al., 2020) in a form of "physics-informed learning" (Kim et al., 2021). During inference, we use a computationally efficient search for the closest matches in the database using a highly compressed image feature representation. We describe an implementation in Pytorch (Paszke et al., 2019) on an Nvidia V100 multi-GPU system.

The remainder of this article is organized as follows: Section 2 describes the novel Double Siamese DL architecture used in this study along with the training process, loss function, and inference process. Section 3 contains details of the experiments that were conducted, describes the computational platform and performance, and quantifies the approximation and prediction error. We analyze the DSNN performance in detail and compare it with alternative architectures. In Section 4, we present comparative visuals of the image retrieval performance on several representative "real-world" examples. We discuss the three

https://doi.org/10.1017/dce.2023.16 Published online by Cambridge University Press

main factors behind the superior performance of our novel architecture. Section 5 presents our conclusions and lists future work.

## 2. Network Architecture

The first CBIR DL architectures were based on convolutional neural networks (Krizhevsky et al., 2012) and autoencoders (Ballard, 1987). While convolution and autoencoders provide a natural mechanism for image dimensionality reduction, single autoencoder networks proved difficult and time-consuming to train, and were limited in image retrieval performance. Wiggers et al. (2019) have shown that Siamese networks perform better than simple CNNs and autoencoders, and we build upon this work. We propose to leverage both physics and "expert knowledge" in a novel "Double Siamese Neural Network" architecture composed of four networks, each using ResNet-18 and a regression module for incorporating physics-informed properties.

In contrast to a conventional Siamese network where only one backbone network is used for image similarity learning, we employ an additional network to leverage both metadata and image similarity. To do the latter, an additional regression term is added to the objective function, similar to multi-task learning (MTL). The purpose of MTL is to train one or multiple shared networks with different tasks for improved prediction performance over networks trained individually with each task. MTL has been used in other CBIR applications, such as in Pisov et al. (2018), where a ResNet-like network is trained with multiple heads for classification and segmentation.

Instead, we opted to follow the approach taken by Standley et al. (2020), by designing a model that can improve our main image retrieval performance with another, possibly conflicting, task of estimating metadata (e.g., porosity and permeability for digital rocks). In this vein, we add another Siamese network for training image similarity along with metadata similarity. Our DSNN's loss function is thus composed of a classification and a regression component (see Figure 1). The classification term determines whether a pair of images is similar and the regression term quantifies the mismatch in the metadata space of the image pair.

### *2.1. Training and the loss function*

To avoid the issue of overfitting while still maintaining high-capacity CNNs, we use transfer learning (Tan et al., 2018; Wiggers et al., 2019) by incorporating two pretrained backbone ResNet-18 networks (He et al., 2016) that were trained with the ImageNet database (Russakovsky et al., 2015). This warm-start of the model parameters helps the model transition from images in the natural domain to digital rock images taken from microscopy or other sensors. We also considered freezing layers of our network, but note superior empirical CBIR recovery using the warm starting approach.

Using similarity labels generated by human experts for a range of image pairs is too costly and work-intensive in our application. We have instead chosen to codify the process by which human experts assess similarity: Visual perception combined with the matching of physical properties is how Earth scientists perform such assessment, and we have mapped this process into our loss function.

The loss function used for training the DSNN has classification and regression terms. As will be described in the next sections in detail, the classification term has an image similarity component based on the images' reduced-dimension feature representation but also a component related to the mismatch of physical properties with additional regression terms in the loss function. Some examples of the approximately 10,000 images used in this study (a small subset of those in our rock image database) are shown in Figure 2. They are acquired through specialized borehole drilling operations and come from various deep subsurface porous rock formations ("reservoirs") across the globe. Millimeter-size pieces of rock ("rock samples") are then cut into slices and subjected to micrography ("thin slices") or scanned and digitized using x-rays ("micro-CT"). We use the rock sample identifier as the label: If a pair of images come from the same sample, they have identical labels. This is because different rock samples from the same reservoir can still exhibit different appearance and properties due to subsurface heterogeneity. The reservoir identifier is used here only during testing, not during the training phase.

https://doi.org/10.1017/dce.2023.16 Published online by Cambridge University Press

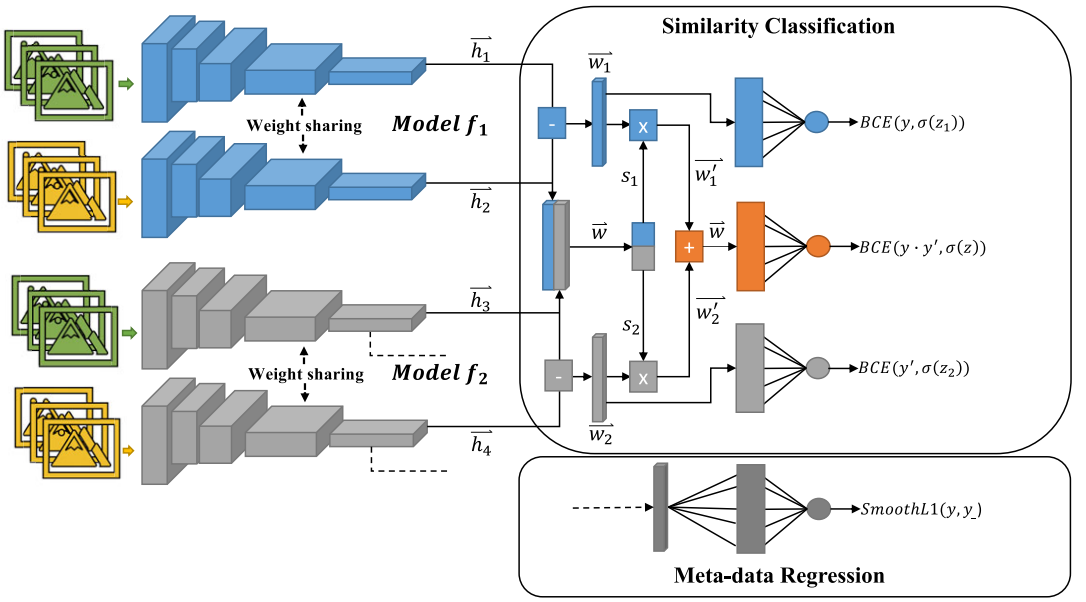


***Figure 1.*** *Proposed Double Siamese Neural Network (DSNN) architecture. The blue Siamese part of the network is trained with image similarity and the gray Siamese part is trained with image and metadata similarity in classification. The dark gray part is trained with metadata for regression. "x," "+," "- "designate multiplication, addition, and subtraction. The green and yellow images are pairs of images, with binary classification label "similar" or "different." After the training phase the networks are used in the inference phase for image retrieval and metadata regression as shown in Figure 3. See text for the mathematical details and symbols.*

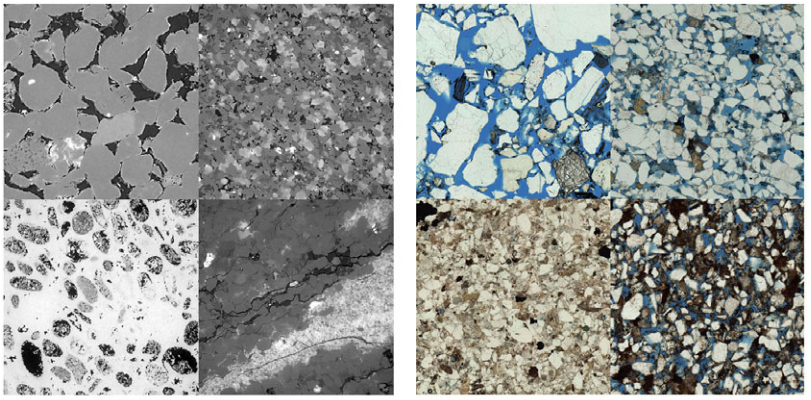


***Figure 2.*** *Examples of 2D micro-CT and thin-section rock images. Micro-CT images are gray scale, while thin-section images are RGB.*

### *2.1.1. Binary classification*

Consider two backbone network models $f_1$ and $f_2$ for feature extraction. The networks $f_1$ and $f_2$ have the same network architecture but the weights are not shared. As noted earlier, we select ImageNet-pretrained ResNet-18s as the initial networks. The ImageNet dataset includes a large number of annotated photographs intended for computer vision research. A pair of images $\vec{A}$ and $\vec{B}$, which are randomly

https://doi.org/10.1017/dce.2023.16 Published online by Cambridge University Press

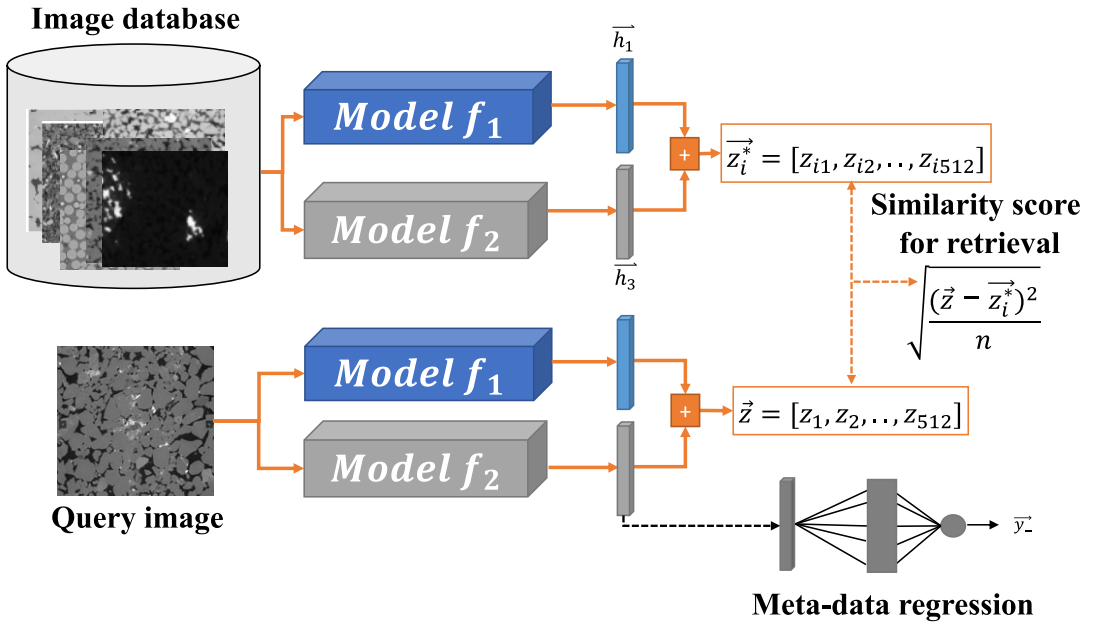


***Figure 3.*** *Image retrieval system with the trained DSNN. Note that the feature encoding for the images in the data base (top) is computed offline and not during inference for a new query image (bottom).*

chosen with 50 % probability from the same or different samples, are passed on to the networks $f_1$ and $f_2$ for feature extraction.

$$\vec{h}_1 = f_1\left(\vec{A}\right), \vec{h}_2 = f_1\left(\vec{B}\right), \quad \vec{h}_3 = f_2\left(\vec{A}\right), \vec{h}_4 = f_2\left(\vec{B}\right). \tag{2.1}$$

After extracting the features from the images (here: a vector of length 512), the difference in latent space is calculated with element-wise absolute difference as follows.

$$\vec{w}_1 = |\vec{h}_1 - \vec{h}_2|, \quad \vec{w}_2 = |\vec{h}_3 - \vec{h}_4| \tag{2.2}$$

$\vec{w}_1$ and $\vec{w}_2$ are then concatenated and the resulting vector $\vec{w}$ passed to a fully connected (FC) layer for gating the outputs, $\vec{w}_1'$ and $\vec{w}_2'$, from each model (see Patel et al., 2017 and Figure 1) for late fusion. The gated outputs are fused with an element-wise mean operation $\left(\vec{w}_1' + \vec{w}_2'\right)/2$ and passed into an output layer, producing a scalar output $z$ as FC($\vec{w}$), which is processed through a sigmoid function $\sigma$ to quantify whether a pair of images $\vec{A}$ and $\vec{B}$ is similar or not. Lastly, each output ($\vec{w}_1$ and $\vec{w}_2$) from the models $f_1$ and $f_2$ is individually passed into a fully connected output layer with a sigmoid nonlinearity and used in constructing the loss function over $N$ image pair samples, $\sigma(z_1)$ and $\sigma(z_2)$, where $z_1$ is FC($\vec{w}_1$), and $z_2$ is FC($\vec{w}_2$).

The loss function in the classification training process employs the standard binary cross entropy (BCE) for image similarity in a Siamese neural network architecture,

$$BCE(y, \hat{y}) = -\frac{1}{N}\sum_{i=0}^{N}\left(y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)\right) \tag{2.3}$$

where $\hat{y} = \sigma(z)$, a given target label $y$ (0 and 1 representing "not similar" and "similar," respectively), and $N$ is the number of samples in the dataset.

We now define two sets of classification labels to compose the overall classification loss. If during training a pair of images is selected from the same rock sample, we define $y = 1$, otherwise $y = 0$. For

https://doi.org/10.1017/dce.2023.16 Published online by Cambridge University Press

metadata similarity labels, if a pair of images $\vec{A}$ and $\vec{B}$ have similar metadata, where the relative difference of the metadata is less than a threshold of 10%, then we define $y' = 1$, otherwise $y' = 0$. Based on these two labeling methods, the classification loss $\mathcal{L}_{\text{clf}}$ is defined as follows:

$$\mathcal{L}_{\text{clf}} = BCE(y \cdot y', \sigma(z)) + BCE(y, \sigma(z_1)) + BCE(y', \sigma(z_2)) \tag{2.4}$$

(the blue, orange, and gray output sections of the network in Figure 1, respectively). The cross term $y \cdot y'$ ensures that images are only counted as "similar" during training when both branches of the double Siamese network classify them as similar.

#### *2.1.2. Metadata regression*

As described earlier, we augment the loss function with a metadata regression term to preferentially retrieve images with similar properties in the inference phase. For our digital rock application, we choose porosity and permeability (see Saxena et al., 2019a). Model $f_2$ is regularized with a regression loss $\mathcal{L}_{\text{reg}}$. A regression module composed of a fully connected layer and an output layer is added in $f_2$, as colored with dark gray in Figure 1. Our approach does not rely on segmentation but predicts rock properties directly from input images via the regression module trained with lab measurements as labels. We use an L1 loss for the metadata regression task $\mathcal{L}_{\text{reg}}$ because it provides the best performance.

The total loss function is a combination of the classification and regression loss:

$$\mathcal{L} = \mathcal{L}_{\text{clf}} + \alpha \mathcal{L}_{\text{reg}} \tag{2.5}$$

Note that the regression term in (2.5) is multiplied with an adjustable scaling factor $\alpha$ to balance the two loss terms so that the magnitude of the classification loss approximately equals the regression loss. Here, we selected a scaling factor $\alpha = 10$. Please see pseudo code of training DSNN in supplementary material section below for details.

### *2.2. Inference*

After training our DSNN, the backbone networks $f_1$ and $f_2$ are used for image retrieval and metadata prediction as shown in Figure 3. Before inference, we first precompute the encoded features from the two trained backbone networks $f_1$ and $f_2$ for all samples in the existing database, including partially labeled samples that were not used for training. When a new query image is presented, its encoded features $\vec{h}_1$ and $\vec{h}_3$ are computed through the two trained backbone networks $f_1$ and $f_2$, respectively. A similarity measure containing both image and metadata terms can be computed for all existing samples in our data base using these encoded features. First, the outputs of both networks, $f_1$ and $f_2$, are fused with an element-wise mean operation as $\vec{z} = \left(\vec{h}_1 + \vec{h}_3\right)/2$. Then, the Euclidean distance is used to compare a query image with the images in the data base (see Figure 3) and retrieve the closest matches. In addition, the output of model $f_2$ is connected with the trained regression module to separately predict metadata for the query image (here: the physical properties porosity and permeability). The DSNN then predicts physical properties in addition to the main CBIR task.

## 3. Results and Discussion

### *3.1. Experimental setup*

We now demonstrate the performance of our physics-informed DSNN architecture on the two datasets of micro-CT and thin-section images, as shown in Figure 2. All calculations were run on a workstation with a 2.50Ghz Intel Xeon® Gold 6248 v80 CPU with 4 Nvidia Tesla® V100-SXM2 GPUs. Our DSNN takes 8 hours for training using 7740 images of 2D micro-CT with 1000 epochs and 12 hours for training with 12240 images of the thin-section dataset with 600 epochs. The most time-consuming part of the training process in every epoch is convolution. The DSNN is implemented with Pytorch 1.7.1 (Paszke et al., 2019). The (pretrained) ResNet-18 (He et al., 2016) is used as a backbone network for DSNN, the last two ResNet layers were removed for feature extraction. The number of trainable parameters is 22.4 million with 26 GB memory usage on each GPU.

https://doi.org/10.1017/dce.2023.16 Published online by Cambridge University Press

The DSNN is trained with a balanced set of selected pairs of images (i.e. half of the pairs are from the same rock samples, see above). We down-sampled all images to $224 \times 224$, randomly flipped images horizontally and vertically with 50% probability for data augmentation, and then normalized with z-scaling. For hyperparameters, we select 1000 epochs for micro-CT and 600 epochs for the thin-section dataset. We set the batch size to 1024 and the learning rate to 0.00001 and 0.0001 for the micro-CT and thin-section dataset, respectively. We use the Adam optimizer to minimize the objective function. To compare with other approaches, we use an ImageNet-pretrained ResNet-34 ("Vanilla ResNet") as the baseline for image retrieval performance. It matches the number of trainable weight parameters of our Double Siamese ResNet-18 network to allow for a fair comparison.

#### *3.1.1. 2D micro-CT image dataset*

We have access to a proprietary database of 3D micro-CT images with image sizes $1000^3$ or $1300^3$, where pixel size ranges from 1.01 μm to 2.12 μm. From these raw samples, 90 2D slices are sampled from each 3D image, which provides a total of 9000 2D slices. 7740 slices are used for training and 1260 slices used for testing the regression performance. To test the image retrieval performance, a separate test set of 1527 2D slices out of 509 3D images is used.

#### *3.1.2. Thin-section Image dataset*

The thin-section image dataset is composed of 264 16-bit RGB raw 2D images of image size $5000 \times 5000$ with pixel size of 1.41 μm. Similar to the 2D micro-CT image dataset, 60 2D slices are extracted from each raw image, for a total of 15840 slices. 12240 images are used for training, and 3600 images for testing the regression performance. For testing image retrieval performance, 37 out-of-sample images are used, a random slice from each of the 37 samples. Porosity is used for regression in this dataset because the permeability is not broadly available.

### *3.2. Image retrieval performance*

For the retrieval performance (estimation error) metric, we use mean average precision at k (mAP@k) (Christopher et al., 2008). It captures how many of the known similar images the network can retrieve on average when a query input image is presented. The total similarity between the query image and the retrievals in the testing phase is quantified with the known "reservoir" designation of the rock samples from which the images were captured (see Section 2). True to the practical application of our method, images are labeled "similar" in the test set if their origin is the same subsurface reservoir. Note that this is different from the DSNN training process where "rock sample" was used as label. Based on this metric, our DSNN outperforms the pretrained ("vanilla") ResNet-34 by 9.8%, the Siamese ResNet-18 by 3.1% and the Siamese ResNet-34 by 1.4% in the 2D micro-CT dataset (see Table 1). In addition to the vanilla ResNet-34, a single Siamese ResNet-18 and a ResNet-34 network was post-trained with our rock images as described above for the DSNN. Comparing with these single Siamese networks, our Double Siamese ResNet-18 shows superior performance.

On the thin-section dataset, our DSNN also outperforms the vanilla ResNet-34 by 22.5%, a Siamese ResNet-18 by 4.3% and a Siamese ResNet-34 by 4.0%. Even though ResNet-34 has more trainable parameters, it does not perform as well as a ResNet-18, which we attribute to overfitting. Generally speaking, we note that all algorithms and models do not perform as well on thin-section data as the micro-CT data due to the lower number of raw samples. Despite that, our DSNN again shows the best image retrieval performance due to the added rock properties.

### *3.3. Regression performance*

As noted earlier, the regression portion of the DSNN network not only improves the image retrieval performance but also enables the prediction of physical properties for a query image. This can happen in two ways: First, the user retrieves the set of best-matching images from the database using the DSNN

https://doi.org/10.1017/dce.2023.16 Published online by Cambridge University Press

***Table 1.*** *Image retrieval performance comparison.*

| Algorithm | mAP@10 | |
|---|---|---|
| | 2D Micro-CT (%) | Thin-section (%) |
| KAZE | 71.7 | 63.5 |
| Vanilla ResNet-34 | 65.1 | 49.8 |
| Siamese ResNet-18 | 71.8 | 68.0 |
| Siamese ResNet-34 with MTL | 72.5 | 69.6 |
| Siamese ResNet-34 | 73.5 | 68.3 |
| Double Siamese ResNet-18 (scratch) | 65.1 | 64.8 |
| Double Siamese ResNet-18 | 74.9 | 72.3 |

*Note.* The KAZE results (Thiele et al., 2020) are listed for reference.

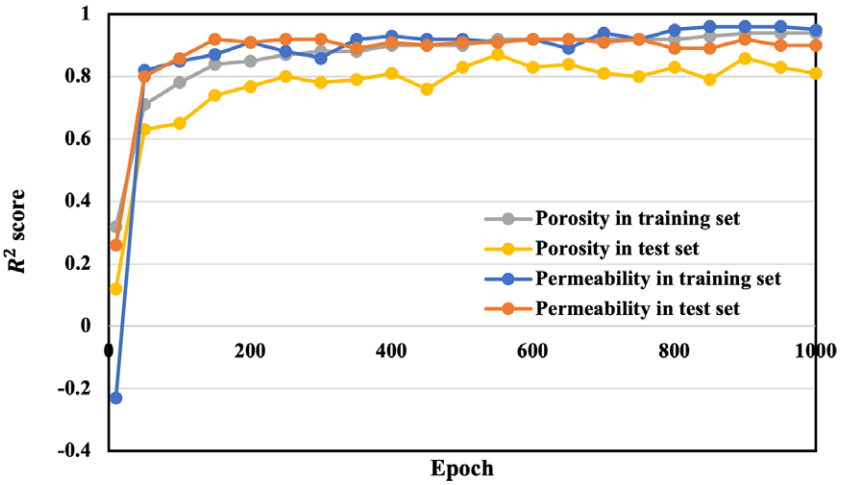


***Figure 4.*** *Regression coefficient of determination $R^2$ for porosity and permeability in the 2D micro-CT dataset.*

and "manually" looks up the physical properties of those. Second, and in a more automated fashion, the trained regression module (Figure 1) can be used directly to predict the physical properties of a query image. The performance of this process (approximation error and estimation error from out-of-sample testing) is shown in Figures 4 and 5. The coefficient of determination $R^2$ is defined as $1-\sum_i(y_i-f_i)^2/\sum_i(y_i-\bar{y})^2$, where $f_i$ and $\bar{y}$ represent prediction and mean of prediction, respectively. Note that only our DSNN has a regression module that can be used in this way, the other neural networks used for comparison in this study cannot predict physical properties. The log of permeability is scaled with a min-max normalization for the training process and permeability itself is then calculated back for testing the regression performance. In terms of rock property prediction accuracy, the $R^2$ for both porosity and permeability, respectively, is 0.8 and 0.9 with our DSNN on the test set after 1000 training epochs.

For the thin-section dataset, we only use porosity in the regression due to the small sample size of available permeability data. Figure 5 shows a final $R^2 = 0.7$ for porosity prediction. This is still considered excellent predictive performance in rock physics and on par with orders-of-magnitude more time-consuming direct computer simulation (Saxena et al., 2017). Our predictive DL approach thus provides a viable alternative to computationally expensive pore-scale flow simulations.

https://doi.org/10.1017/dce.2023.16 Published online by Cambridge University Press

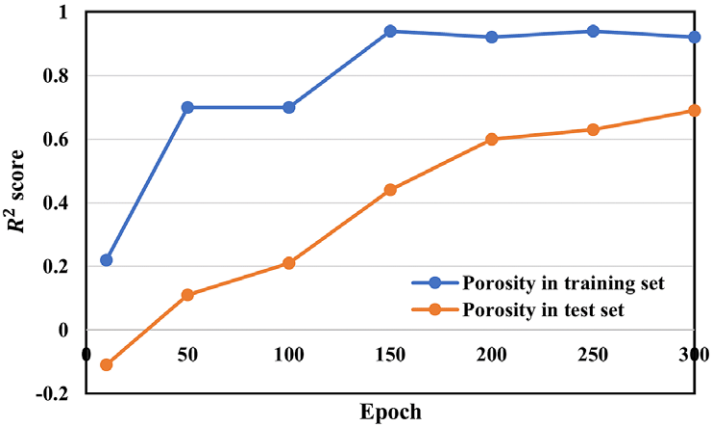


***Figure 5.*** *Regression coefficient of determination $R^2$ for porosity in the thin-section dataset.*

### *3.4. Quantitative image retrieval analysis*

In this section, we assess the DSNN performance for *rock property* predictions. We compare query images with their best-matching three retrievals in all samples of a separate test set. This is of paramount importance for digital rocks because users often infer rock properties by studying "similar" rocks returned from the CBIR query.

For the 2D micro-CT dataset, porosity and permeability of a query image and the top three retrievals are compared in Figure 6, with a mean and a standard deviation of (0.21, 0.07) for the porosity and (1.52, 1.01) for permeability. For a high-performing network all top three retrievals should lie on the diagonal line. From Figure 6a it is apparent that the vanilla ResNet-34 does not perform as well as our DSNN. This is because it was not trained on rock images. The average perpendicular distance of the top-1,2,3 retrievals to the diagonal line is 0.0043 from the vanilla ResNet, and 0.002 from our DSNN, about a two times better retrieval quality in terms of similar porosity. Figure 6b shows that our DSNN retrieves most of the images in each sample correctly. The average perpendicular distance from the diagonal of the top-1,2,3 retrievals for permeability is 2.17 from the vanilla ResNet, and 0.42 from our DSNN, about five times better in terms of similar permeability. While some images are not retrieved from the same sample, the porosity differences between the query and the retrievals are small. Due to the rock property similarity-based labels and the regularization with the property regression module, the image retrieval quality goes up. In Figure 7, the improved retrieval performance of our algorithm is shown in histogram form where the horizontal axis represents metadata difference between query image and its top three retrievals. Our DSNN retrieves rock samples which have better matching porosity and permeability than the vanilla ResNet-34, as demonstrated by higher peaks near zero and lower peaks away from zero. The permeability recovery performance for DSNN is also shown in Figure 7.

The thin-section image results are shown in Figure 8. Only porosity is used as physical quantity with (0.26, 0.06) as mean and a standard deviation. Figure 8a shows that the vanilla ResNet-34 is unable to consistently retrieve samples with similar porosity. More than half of the samples are far off the diagonal line. Our DSNN (Figure 8b) reliably retrieves rock samples with similar porosity. Figure 8c,d demonstrate that the DSNN retrieves more similar rock samples with a higher peak at zero compared with the vanilla ResNet-34. In terms of the average perpendicular distance metric, the vanilla ResNet shows 0.03, and our DSNN still shows improvements as 0.02.

## 4. Practical Examples with Discussion

Figures 9–12 show examples of actual (both Micro-CT and thin-section) rock images retrieved from the Shell image database using both a vanilla ResNet-34 and our DSNN. A comparison demonstrates the

https://doi.org/10.1017/dce.2023.16 Published online by Cambridge University Press

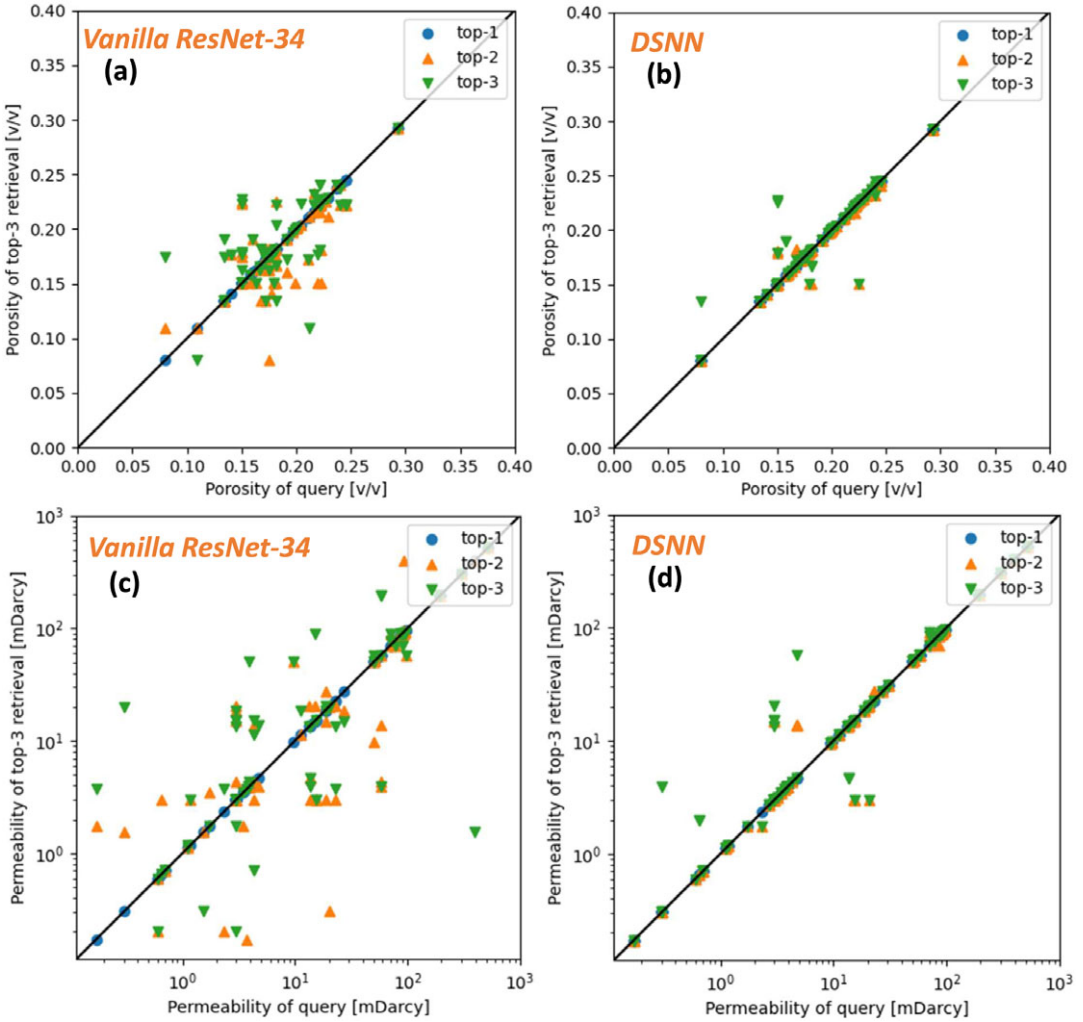


***Figure 6.*** *Cross plot of porosity and permeability of query images and their top three matching retrieval images for the 2D micro-CT dataset. (a and b) are for porosity and (c and d) are for permeability. Our DSNN (right panels) reliably retrieves rock samples with similar rock properties; most of the retrievals are on the diagonal line. The top, second-, and third-best matches are colored blue, orange, and green, respectively.*

superior performance of our DSNN architecture and training procedure. For the micro-CT dataset, in Figures 9 and 10, the first row (red box) contains the top three ranking images retrieved from the same rock sample as the query image. The ranking metric is the L2 distance of the feature vectors described in Section 2.2. The second row (green box) includes the top three ranked retrievals from the same geological subsurface rock formation ("reservoir") as the query but not from the same physical rock sample. The third row (blue box) shows the top three ranked retrievals from images of rock samples that originate in different reservoirs than the query image. This similarity assessment is used in practice to identify geologically similar rocks already in the database that can serve as useful analogs with known properties. In these figures, porosity is abbreviated as *por*, permeability as *perm*, and absolute retrieval ranking as *ABS ranking*. Note that the rock properties of many images in the database were not measured, as indicated by *NaN*.

https://doi.org/10.1017/dce.2023.16 Published online by Cambridge University Press

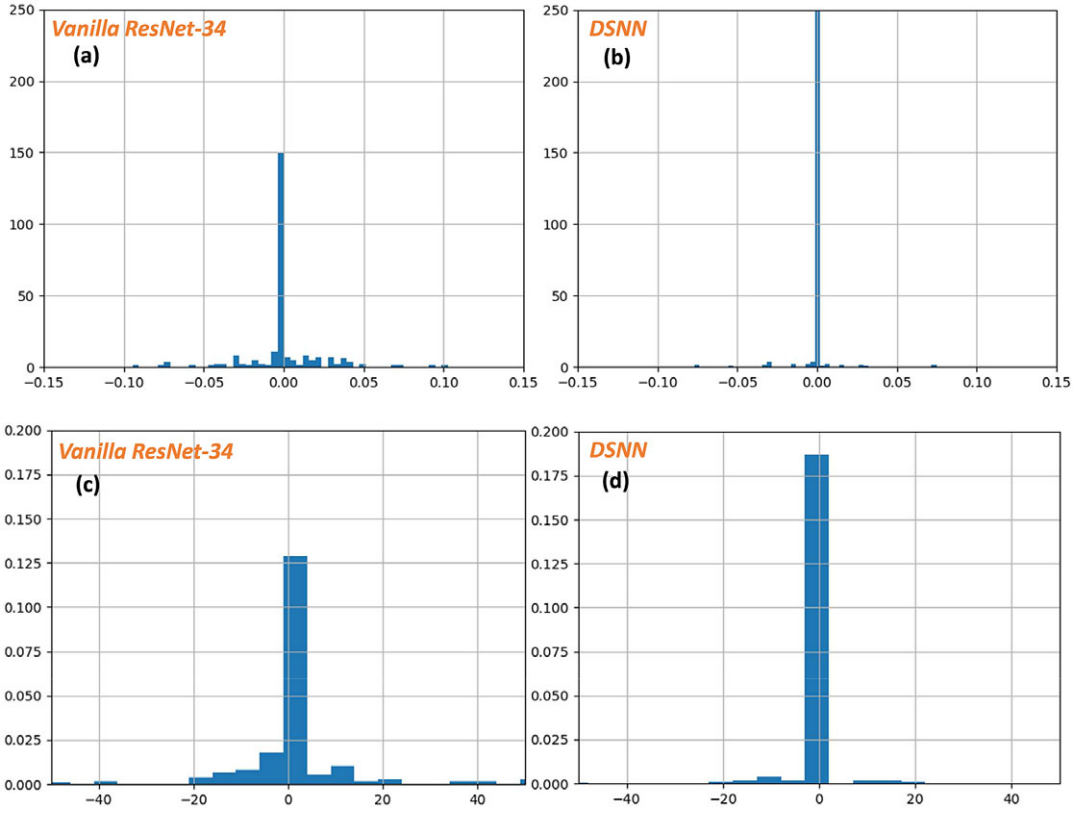


***Figure 7.*** *(Porosity, permeability) of query and (porosity, permeability) of top three retrievals compared for the 2D micro-CT dataset. (a and b) are for porosity and (c and d) are for permeability. The error histogram for our DSNN shows a narrower spread around zero (perfect match) than the vanilla ResNet-34.*

In Figure 9, one slice of a rock sample from reservoir "E2" is used as query (top left in the first row). Ideally, the images retrieved from the same rock sample as the query image should be among the top-ranking images, and the query image itself should be the highest-ranked result. Note that the images retrieved by the vanilla ResNet-34 from the same rock sample are the same ones as those retrieved by our DSNN but have lower absolute ranking, an undesirable discrepancy. The secondand third-ranking images retrieved by the vanilla ResNet-34 are from a different rock sample; the two retrieved images are also visually dissimilar to the query image and belong to different rock types defined by distinct rock property trends. That is despite the fact that the database included multiple 2D slices from the same rock sample from which the query image was selected. It is apparent that a pretrained vanilla ResNet-34 does not function well for retrieving similar images in the materials domain.

On the other hand, our DSNN successfully retrieves the secondand third-ranked images from the same rock sample as the query image. Our DSNN also shows stable regression performance on porosity *por'* and permeability *perm'*: A 10% error in porosity and log(permeability) is considered satisfactory in practice (Andrä et al., 2013; Woods, 2014; Saxena et al., 2017). The blue box (third row; top three ranked retrievals from different reservoirs) is a tool to assess which other reservoir rocks can be used as analogs for the query rock sample. Rocks of similar geologic origin can be expected to have similar rock properties. Reservoirs E2 (query image), E1, and Z are from the same geological formation but were penetrated by different wells several kilometers away from reservoir E2. The three reservoirs are located at different depths. Reservoirs E2, E1, and Z have seen the same degree of rock compaction over geologic

https://doi.org/10.1017/dce.2023.16 Published online by Cambridge University Press

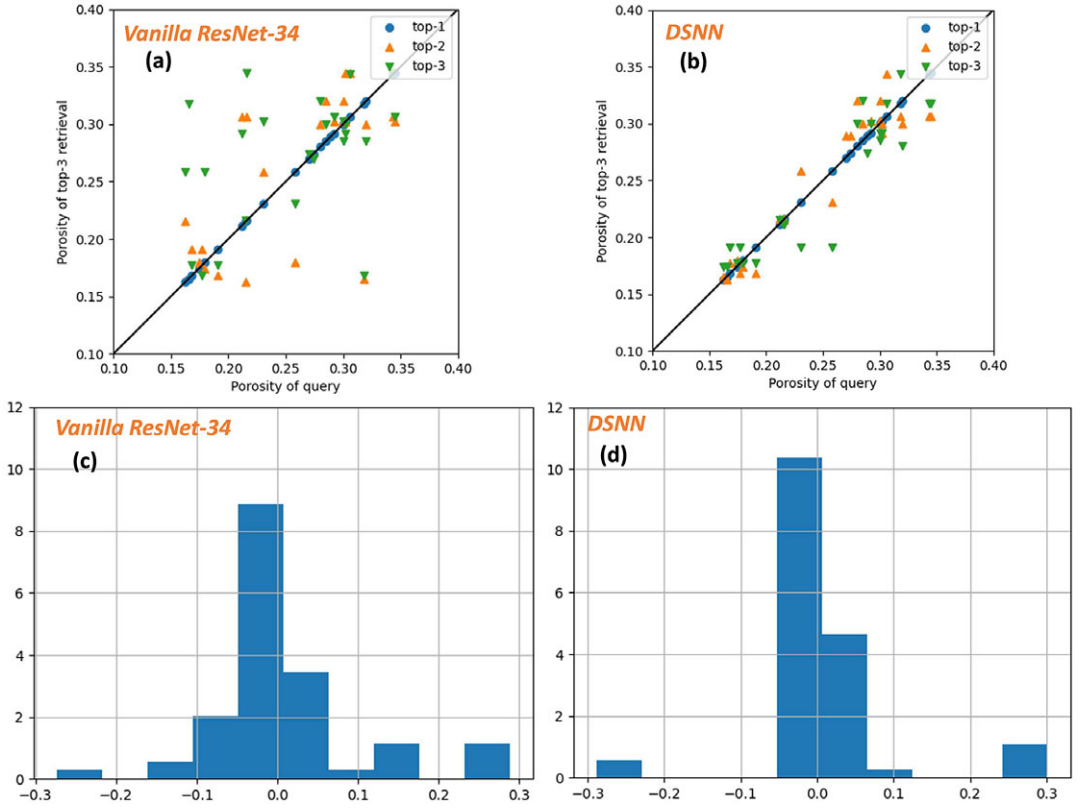


***Figure 8.*** *Comparison of the porosity of a query image and the top three ranked images retrieved in the thin-section dataset. (a) and (b) are cross plots of porosity of query images and the top3 matching retrieval images. (c) and (d) are histograms to display the cross plots in terms of distributions. Similar to the results in the 2D micro-CT dataset, our DSNN in (d) has a higher peak at zero.*

times, and are known to have similar rock properties and trends. Note how all images retrieved by our DSNN are visually similar to the query image. Our DSNN correctly predicts reservoir E1 as a good analog for reservoir E2. It also predicts reservoir Z as the next best analog for reservoir E2. The retrievals from the vanilla ResNet-34 are not satisfactory since the top retrieved image from a different reservoir (blue box) is from reservoir C which is from a very different geological formation. The next two predictions from the vanilla ResNet-34, in the blue box, are from reservoir E1 which is the same geological formation as reservoir E2 but is known to be less compacted and of higher porosity. Note also how multiple retrieved images are visually dissimilar to the query image.

Figure 10 shows the same results as Figure 9 but for a different query image obtained from a rock sample from reservoir "K". The key difference between the results predicted by the two networks is in the prediction of similar images from different reservoirs (bottom blue box in Figure 10). Results from the vanilla ResNet-34 are not satisfactory for two important reasons: Firstly, for the query image from reservoir K, the retrieved images from reservoirs R and T have significantly higher concentration of "heavy" minerals (brighter pixels in the images) and are visually dissimilar to the query image. Secondly, reservoirs R and T have different geologic origins to reservoir K. Reservoirs R and T are aeolian deposits, whereas reservoir K has a turbitic origin. Our DSNN predicts reservoir Z as a good analog for reservoir K; both of these reservoirs are turbidite deposits and are correctly predicted to have similar porosity and permeability.

The thin-section dataset results are shown in Figures 11 and 12. Only one thin-section sample was analyzed for the purpose of our discussion here so that only two rows of images are shown. The first row

https://doi.org/10.1017/dce.2023.16 Published online by Cambridge University Press

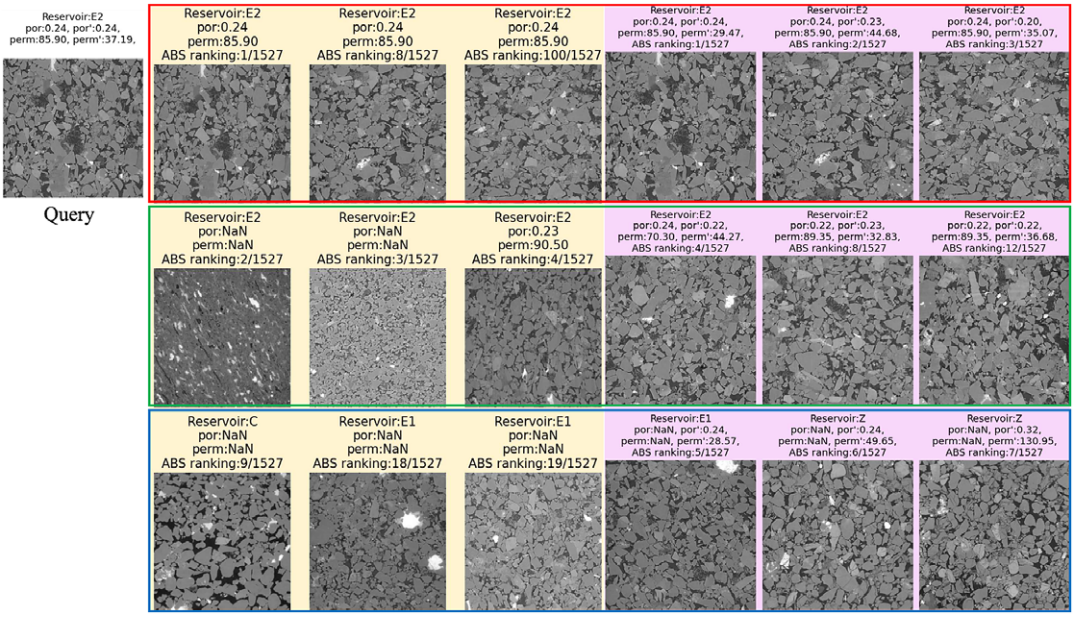


***Figure 9.*** *Qualitative retrieval results with reservoir E2 on 2D micro-CT dataset. Vanilla ResNet-34 (yellow) vs our DSNN (purple). por and perm are lab-measured porosity and permeability. por′ and perm′ represent predicted porosity and permeability from DSNN.*

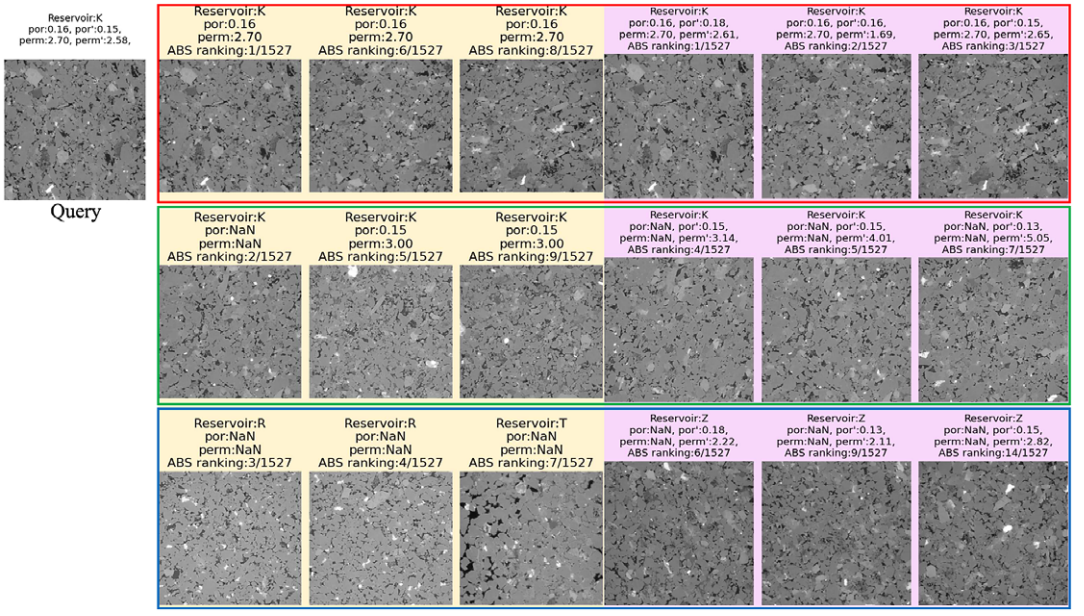


***Figure 10.*** *Qualitative retrieval results with Reservoir K on 2D micro-CT dataset. Vanilla ResNet-34 (yellow) vs our DSNN (purple).*

(green box) represents the top three retrievals from the same reservoir. The bottom row (blue box) includes the top three retrievals from different reservoirs in our database. One slice of reservoir "U" is used as query (Figure 11). Based on the absolute ranking from the retrieval, both the vanilla ResNet-34 and our DSNN

https://doi.org/10.1017/dce.2023.16 Published online by Cambridge University Press

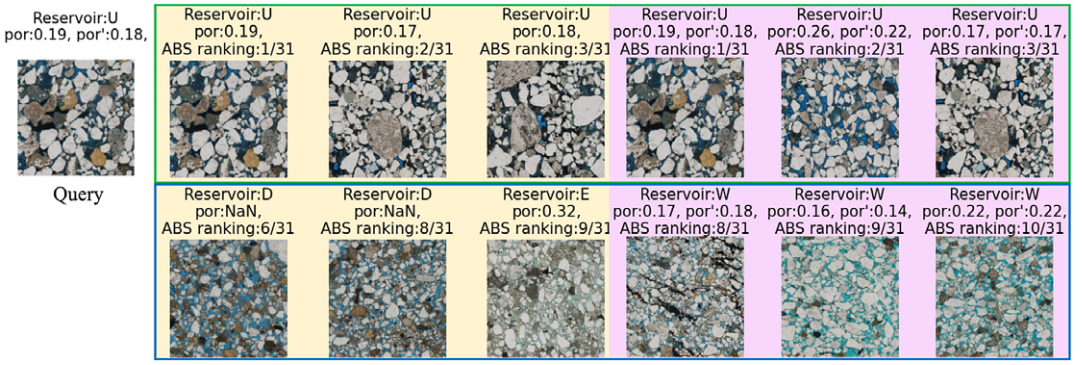


***Figure 11.*** *Qualitative retrieval results with reservoir U on the 2D thin-section dataset. Vanilla ResNet-34 (yellow) vs our DSNN (purple). por is a lab-measured porosity and por′ represents porosity predicted from DSNN.*

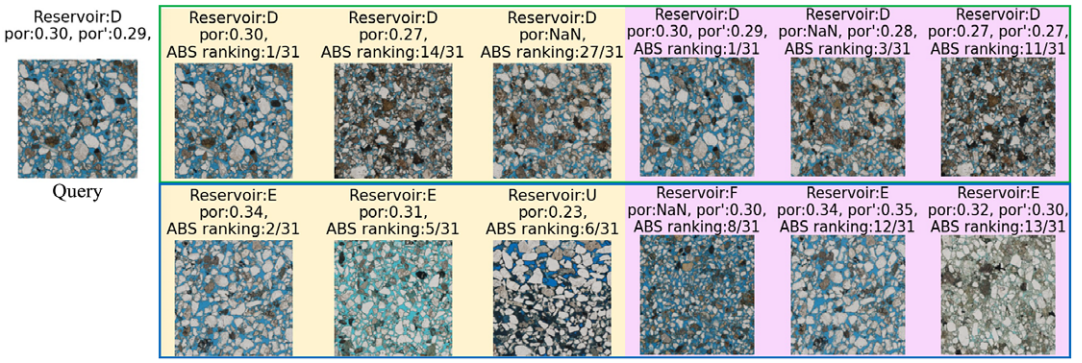


***Figure 12.*** *Qualitative retrieval results for reservoir D on the 2D thin-section dataset. Vanilla ResNet-34 (yellow) vs our DSNN (purple).*

generally retrieve similar rock samples from the same reservoir. The key difference between the two networks is seen in the blue box (second row; top three retrievals from different reservoirs): The vanilla ResNet-34 predicts rocks from reservoir D as the top analogs, whereas our DSNN predicts reservoir W as the closest analog for reservoir rock U. Reservoir W (but not reservoir D) is known to have a similar degree of rock compaction and range of rock properties as reservoir U. This confirms that our DSNN is able to capture attributes that identify petrophysically similar rocks from different reservoir. The retrieval performance is improved by adding a second Siamese network with a regression module and by training with both image and metadata simultaneously.

Figure 12 shows the same analysis as in Figure 11 but for a query image from reservoir D. The vanilla ResNet-34 predicts reservoir U as one of the top three images retrieved from different reservoirs (blue box). However, the two rock images from reservoirs D and U are not only different in porosity but are also dissimilar in texture and mineralogy. The images retrieved from our DSNN are consistent with the known rock physics.

## 5. Conclusions

We propose a novel double Siamese network for content-based image retrieval of industrial materials images in the form of "digital rocks." The performance-limiting challenge is the low number of available labeled images for supervised learning. To still maintain high-capacity networks without overfitting, we

https://doi.org/10.1017/dce.2023.16 Published online by Cambridge University Press

leverage pretrained ResNet-18 models to warm-start the training process. We also augment the training loss by incorporating prior knowledge in the form of physical parameters, that is, porosity and permeability. We demonstrate superior empirical performance for image similarity through the domain standard map@k metric. Our approach also allows for estimation of metadata from images, and we observe excellent predictive power in terms of $R^2$. We present a sample of qualitative search results, demonstrating the ability of our DSNN to provide relevant and impactful image retrieval results. Future work includes an extension to full 3D, that is, to using 3D images of rock samples and using 3D kernels in the neural network. Permeability prediction could thus potentially be improved because its volumetric nature cannot be captured completely with 2D kernels. Our novel approach has the potential to replace direct computer simulation of physical properties in digital industrial materials images with orders-of-magnitude faster direct prediction.

Pseudo code of our proposed DSNN on the training process is shown below.

**Algorithm 1** Pseudocode for training the double Siamese Neural Network. See Figure 1 for the diagrammatic representation.

**INPUT:** Pairs of input images $(X_A, X_B)$, image similarity-based label y, metadata similarity-based label y', sigmoid $\sigma$, pairs of metadata label $(y_{metaA}, y_{metaB})$, loss balancing term $\alpha = 10$, number of iterations i

**for** 1 to i **do**

$\vec{h}_1 = f_1(X_A), \vec{h}_2 = f_1(X_B)$ ▷ Feature extraction via backbone network $f_1$

$\vec{h}_3 = f_2(X_A), \vec{h}_4 = f_2(X_B)$ ▷ Feature extraction via backbone network $f_2$

$\vec{w}_1 = |\vec{h}_1 - \vec{h}_2|, \vec{w}_2 = |\vec{h}_3 - \vec{h}_4|$ ▷ Calculate feature absolute difference

$\vec{w} = softmax\left(\vec{w}_1 \| \vec{w}_2\right)$ ▷ Concatenate two vectors to generate two scalars $s_1, s_2$

$\vec{w}'_1 = s_1 \cdot \vec{w}_1, \vec{w}'_2 = s_2 \cdot \vec{w}_2$ ▷ Gating feature vectors

$z = FC\left(\left(\vec{w}'_1 + \vec{w}'_2\right)/2\right)$ ▷ A final output with late fusion

$z_1 = FC\left(\vec{w}_1\right)$ ▷ A final output from model $f_1$

$z_2 = FC\left(\vec{w}_2\right)$ ▷ A final output from model $f_2$

$\mathcal{L}_{\text{clf}} = BCE(y, \sigma(z_1)) + BCE(y \cdot y', \sigma(z)) + BCE(y', \sigma(z_2))$ ▷ Apply BCE loss for classification

$\mathcal{L}_{\text{reg}} = SmoothL1\left(y_{metaA}, \vec{h}_3\right) + SmoothL1\left(y_{metaB}, \vec{h}_4\right)$ ▷ Apply SmoothL1 loss for regression

$\mathcal{L} = \mathcal{L}_{\text{clf}} + \alpha \mathcal{L}_{\text{reg}}$ ▷ Total loss function for backpropagation

**end for**

**Acknowledgments.** We thank Ronny Hofmann, Dipanwita Nandy, Rishu Saxena, and Lalit Gupta for helpful discussions, and Shell for granting permission to publish this work.

**Author contribution.** Conceptualization and investigation: M.S.S. and C.T.; Visualization and writing original draft: M.S.S.; Methodology, software, and validation: M.S.S., J.V., N.S., and D.H.; Project administration, resources, and supervision: J.V., N.S., and D.H.; Reviewing and editing: M.S.S., C.T., J.V., N.S., and D.H.; All authors approved the final manuscript.

**Competing interest.** The authors declare none.

**Data availability statement.** Source codes are uploaded on this url: https://github.com/sede-open/RockIR.

**Funding statement.** This research was funded by Shell International Exploration and Production (US) Inc.

## References

**Ahuja VR**, **Gupta U**, **Rapole SR**, *et al.* (2022) Siamese-sr: A siamese super-resolution model for boosting resolution of digital rock images for improved petrophysical property estimation. *IEEE Transactions on Image Processing 31*. https://doi.org/10.1109/TIP.2022.3172211

**Alcantarilla PF**, **Bartoli A and Davison AJ** (2012) Kaze features. In *European Conference on Computer Vision*. New York City, USA: Springer, pp. 214–227.

**Al-Marzouqi H** (2018) Digital rock physics: Using ct scans to compute rock properties. *IEEE Signal Processing Magazine 35*(2), 121–131. https://doi.org/10.1109/MSP.2017.2784459

**Andrä H**, **Combaret N**, **Dvorkin J**, **Glatt E**, **Han J**, **Kabel M**, **Keehm Y**, **Krzikalla F**, **Lee M**, **Madonna C**, **Marsh M**. (2013) Digital rock physics benchmarks—Part ii: Computing effective properties. *Computers & Geosciences 50*, 33–43.

**Araya-Polo M**, **Alpak FO**, **Hunter S**, **Hofmann R and Saxena N** (2018) Deep learning-driven pore-scale simulation for permeability estimation. In *ECMOR XVI - 16th European Conference on the Mathematics of Oil Recovery*, Vol. *2018*, No. 1, pp. 1–14. European Association of Geoscientists & Engineers.

**Ballard DH** (1987) Modular learning in neural networks. In *AAAI-87 Proceedings*, Vol. *647*, pp. 279–284.

**Bell S**, **Upchurch P**, **Snavely N and Bala K** (2015) Material recognition in the wild with the materials in context database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3479–3487.

**Blunt MJ**, **Jackson MD**, **Piri M and Valvatne PH** (2002) Detailed physics, predictive capabilities and macroscopic consequences for pore-network models of multiphase flow. *Advances in Water Resources 25*(8-12), 1069–1089.

**Bromley J**, **Guyon I**, **LeCun Y**, **Säckinger E and Shah R** (1993) Signature verification using a "siamese" time delay neural network. In *Proceedings of the 6th International Conference on Neural Information Processing Systems*, ser. NIPS'93, Denver, Colorado. Burlington, MA: Morgan Kaufmann Publishers Inc., pp. 737–744.

**Chen W**, **Liu Y**, **Wang W**, **Bakker EM**, **Georgiou T**, **Fieguth P**, **Liu L**, **Lew MS**. (2022) Deep Learning for Instance Retrieval: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 7270–7292

**Chicco D** (2021) Siamese neural networks: An overview. In Cartwright H (ed.) *Artificial Neural Networks*. New York, NY: Springer US, pp. 73–94. https://doi.org/10.1007/978-1-0716-0826-5_3

**Christopher HS**, **Manning D and Raghavan P** (2008) *Introduction to Information Retrieval*. Cambridge: Cambridge University Press.

**Devareddi RB and Srikrishna A** (2022) Review on content-based image retrieval models for efficient feature extraction for data analysis. In *International Conference on Electronics and Renewable Systems (ICEARS)*. New York City, USA: IEEE, pp. 969–980.

**Dubey SR** (2022) A decade survey of content based image retrieval using deep learning. *IEEE Transactions on Circuits and Systems for Video Technology 32*(5), 2687–2704. https://doi.org/10.1109/TCSVT.2021.3080920

**Gudivada VN and Raghavan VV** (1995) Content based image retrieval systems. *Computer 28*(9), 18–22.

**He K**, **Zhang X**, **Ren S and Sun J** (2016) Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778.

**Kim S**, **Kim I**, **Lee J and Lee S** (2021) Knowledge integration into deep learning in dynamical systems: An overview and taxonomy. *Journal of Mechanical Science and Technology 35*, 1331–1342. https://doi.org/10.1007/s12206-021-0342-5

**Krizhevsky A**, **Sutskever I and Hinton GE** (2012) Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems 25*, pp. 84–90.

**Kulkarni S and Manu T** (2022) Content based image retrieval: A literature review. In *2022 6th International Conference on Intelligent Computing and Control Systems (ICICCS)*. New York City, USA: IEEE, pp. 1580–1587.

**Paszke A**, **Gross S**, **Massa F**, **Lerer A**, **Bradbury J**, **Chanan G**, **Killeen T**, **Lin Z**, **Gimelshein N**, **Antiga L**, **Desmaison A**. (2019) Pytorch: An imperative style, high-performance deep learning library. In Wallach H, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox E and Garnett R (eds), *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., pp. 8024–8035. Available at http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

**Patel N**, **Choromanska A**, **Krishnamurthy P and Khorrami F** (2017) Sensor modality fusion with CNNs for UGV autonomous driving in indoor environments. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. New York City, USA: IEEE, pp. 1531–1536.

**Pisov M**, **Makarchuk G**, **Kostjuchenko V**, **Dalechina A**, **Golanov A and Belyaev M** (2018) Brain tumor image retrieval via multitask learning. Preprint, arXiv:1810.09369.

**Pradhan J**, **Pal AK and Banka H** (2021) Medical image retrieval system using deep learning techniques. In Elloumi M (ed.), *Deep Learning for Biomedical Data Analysis: Techniques, Approaches, and Applications*. Cham: Springer International Publishing, pp. 101–128. https://doi.org/10.1007/978-3-030-71676-9_5

**Rian Z**, **Christanti V and Hendryli J** (2019) Content-based image retrieval using convolutional neural networks. In *2019 IEEE International Conference on Signals and Systems (ICSigSys)*, pp. 1–7. https://doi.org/10.1109/ICSIGSYS.2019.8811089

**Russakovsky O**, **Deng J**, **Su H**, *et al.* (2015) ImageNet large scale visual recognition challenge. *International Journal of Computer Vision (IJCV) 115*(3), 211–252. https://doi.org/10.1007/s11263-015-0816-y

**Saxena N**, **Day-Stirrat RJ**, **Hows A and Hofmann R** (2021) Application of deep learning for semantic segmentation of sandstone thin sections. *Computers & Geosciences 152*, 104778.

**Saxena N**, **Hofmann R**, **Alpak FO**, **Berg S**, **Dietderich J**, **Agarwal U**, **Tandon K**, **Hunter S**, **Freeman J**, **Wilson OB**. (2017) References and benchmarks for pore-scale flow simulated using micro-ct images of porous media and digital rocks. *Advances in Water Resources 109*, 211–235.

**Saxena N**, **Hows A**, **Hofmann R**, **Freeman J and Appel M** (2019a) Estimating pore volume of rocks from pore-scale imaging. *Transport in Porous Media 129*(1), 403–412.

https://doi.org/10.1017/dce.2023.16 Published online by Cambridge University Press

**Saxena N**, **Hows A**, **Hofmann R**, **Alpak FO**, **Dietderich J**, **Appel M**, **Freeman J**, **De Jong H** (2019b) Rock properties from micro-ct images: Digital rock transforms for resolution, pore volume, and field of view. *Advances in Water Resources 134*, 103419.

**Simran A**, **Kumar PS and Bachu S** (2021) Content based image retrieval using deep learning convolutional neural network. *IOP Conference Series: Materials Science and Engineering 1084*(1), 012 026. https://doi.org/10.1088/1757-899x/1084/1/012026

**Standley T**, **Zamir A**, **Chen D**, **Guibas L**, **Malik J and Savarese S** (2020) Which tasks should be learned together in multi-task learning? In *International Conference on Machine Learning*. Cambridge, University Printing House Shaftesbury Road, United Kingdom: PMLR, pp. 9120–9132.

**Tan C**, **Sun F**, **Kong T**, **Zhang W**, **Yang C and Liu C** (2018) A survey on deep transfer learning. In *International Conference on Artificial Neural Networks*. New York City, USA: Springer, pp. 270–279.

**Thiele C**, **Saxena N and Hohl D** (2020) Content based image retrieval (cbir) using deep neural networks. In *Matlab Energy Conference 2020*. Available at https://www.mathworks.com/company/events/conferences/matlab-energyconference/2020/proceedings.html.

**Voulodimos A**, **Doulamis N**, **Doulamis A and Protopapadakis E** (2018) Deep learning for computer vision: A brief review. *Computational Intelligence and Neuroscience 2018*.

**Wang Y**, **Yao Q**, **Kwok JT and Ni LM** (2020) Generalizing from a few examples: A survey on few-shot learning. *ACM Computing Surveys (csur) 53*(3), 1–34.

**Wiggers KL**, **Britto AS**, **Heutte L**, **Koerich AL and Oliveira LS** (2019) Image retrieval and pattern spotting using siamese neural network. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8. https://doi.org/10.1109/IJCNN.2019.8852197

**Woods AW** (2014) Accounting for uncertainty. In *Flowin Porous Rocks: Energy and Environmental Applications*. Cambridge: Cambridge University Press, pp. 52–69. https://doi.org/10.1017/CBO9781107588677.005

**Zagoruyko S and Komodakis N** (2015) Learning to compare image patches via convolutional neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*. New York City, USA: IEEE Computer Society, pp. 4353–4361. https://doi.org/10.1109/CVPR.2015.7299064

**Zheng L**, **Yang Y and Tian Q** (2018) Sift meets cnn: A decade survey of instance retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence 40*(5), 1224–1244. https://doi.org/10.1109/TPAMI.2017.2709749

**Cite this article: Shim MS, Thiele C, Vila J, Saxena N and Hohl D** (2023). Content-based image retrieval for industrial material images with deep learning and encoded physical properties. *Data-Centric Engineering*, 4, e21. doi:10.1017/dce.2023.16