



# Impact of dataset size and convolutional neural network architecture on transfer learning for carbonate rock classification

Harriet L. Dawson<sup>\*</sup>, Olivier Dubrule, Cédric M. John

Department of Earth Science and Engineering, Imperial College, Prince Consort Road, London, SW7 2BP, United Kingdom

## ARTICLE INFO

### Keywords:

Deep learning  
Machine learning  
Dunham classification  
Geological images

## ABSTRACT

Modern geological practices, in both industry and academia, rely largely on a legacy of observational data at a range of scales. However, widespread ambiguities in the petrographic description of rock facies reduce the reliability of descriptive data. Previous studies have demonstrated a great potential for the use of convolutional neural networks (CNNs) in the classification of facies from digital images; however, it remains to be determined which of the available CNN architectures performs best for a geological classification task. We evaluate the ability of top-performing CNNs to classify carbonate core images using transfer learning, systematically developing a performance comparison between these architectures on a complex geological dataset. Three datasets with orders of magnitude difference in data quantity (7000–104,000 samples) were created that contain images across seven classes from the modified Dunham Classification for carbonate rocks. Following training of nine different CNNs of four architectures on these datasets, we find the Inception-v3 architecture to be most suited to this classification task, achieving 92% accuracy when trained on the larger dataset. Furthermore, we show that even when using transfer learning the size of the dataset plays a key role in the performance of the models, with those trained on the smaller datasets showing a strong tendency to overfit. This has direct implications for the application of deep learning in geosciences as many papers currently published use very small datasets of less than 5000 samples. Application of the framework developed in this research could aid the future of deep learning based carbonate classification, with further potential to be easily modified to suit the classification of cores originating from different formations and lithologies.

## 1. Introduction

Descriptions of observed geological features play a large role in driving applications and research within the Earth Sciences; for example, when facies descriptions of core data are used to derive regional stratigraphic trends, or as the basis for petrophysical classifications (Hull et al., 2015; John and Kanagandran, 2019). The modified Dunham classification (Dunham, 1962; Embry and Klován, 1971) is acknowledged as one of the most commonly used classification schemes for the systematic description of carbonate rocks. However, recent studies have shown that even with the well-established and clearly defined divisions of the scheme, experienced sedimentologists may often classify alike facies using different textural names (Lokier and Al Junaibi, 2016). Deep learning presents a data-driven approach to deriving predictive models from observational data.

The use of machine learning and deep learning in geoscience is rapidly growing; however, progression has been relatively uneven and,

despite increasing popularity within the geoscientific community, practitioners have been comparatively slow to engage with these recent advancements (Bergen et al., 2019; Reichstein et al., 2019). Previous classification tasks have focused largely on interpreting data from seismic or from wireline logs (e.g. Hall, 2016; Qian et al., 2018; Saporetti et al., 2018; Halotel et al., 2020). A pilot study in our research group (John and Kanagandran, 2019) tested the effectiveness of neural networks in recognising carbonate rock facies according to the Dunham classification scheme. Using high-resolution core images from Ocean Drilling Program (ODP) Leg 194, the convolutional neural network (CNN) model achieved 89.2% accuracy across seven classes. The results from this study show great potential for the use of deep learning in carbonate classification from digital images. Several studies now have applied CNN models to broad lithofacies classifications in both core and thin section images (Baraboshkin et al., 2020; Pires de Lima et al., 2020; Koeshidayatullah et al., 2020; Chawshin et al., 2021). However, it remains to be determined which of the available CNN architectures

<sup>\*</sup> Corresponding author.

E-mail address: [h.dawson19@imperial.ac.uk](mailto:h.dawson19@imperial.ac.uk) (H.L. Dawson).

<https://doi.org/10.1016/j.cageo.2022.105284>

Received 22 June 2022; Received in revised form 4 October 2022; Accepted 7 December 2022

Available online 8 December 2022

0098-3004/© 2022 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

performs best for the classification of carbonate core images.

Herein, this paper evaluates the ability of a selection of the top performing CNNs to classify carbonate core images, systematically developing a performance comparison between these architectures on a complex geological dataset. This deep learning approach is applied to three datasets with orders of magnitude difference in data quantity (a 104k dataset, a 42k dataset, and a 7k dataset), highlighting the impact of dataset size on the performance and reliable application of the models. The findings of this research serve as a necessary first step for further research into training deep learning algorithms for carbonate textural classifications. Application of these advanced digital technologies has the potential to revolutionise descriptive geology, with more accurate and uniform facies descriptions, faster analysis times, and reduction of natural biases.

### 1.1. Background

Machine learning has been successfully applied to various sub-disciplines of the Earth Sciences for over three decades (e.g., Baldwin et al., 1990; Brown et al., 2000; de Matos et al., 2007; Pires de Lima et al., 2019). Current deep learning applications within the geoscientific community include, but are not limited to, seismic facies classification (e.g. West et al., 2002; Chevitere et al., 2018), lithofacies classification from wireline logs (e.g. Bestagini et al., 2017; Halotel et al., 2020), volcanic ash detection (e.g. Shoji et al., 2018; Torrisi et al., 2021), seismology (Kortström et al., 2016; Mousavi et al., 2019), and inverse problems (e.g. Mosser et al., 2020).

Despite these successes, many existing datasets from long-established sources remain largely unexplored; partly due to the lack of properly labelled data and due to the inefficiency of traditional analytical techniques, which are typically resource intensive (Oikonomou et al., 2017). One of the key advantages of using deep learning for geoscience is the ability to process large volumes of multidimensional data, making these analytical techniques ultimately more time and cost effective. It is anticipated that the application of machine learning will see dramatic progress in the automation of complex prediction tasks, the solution of inverse problems through multiscale modelling, and the discovery of new or alternative patterns and relationships that are not readily visible to humans (Karpatne et al., 2019; Mosser et al., 2020). Recent efforts to promote the use of benchmark geoscientific datasets in global and local competitions have the potential to drive deeper, broader, and more collaborative efforts e.g. the SEG Machine Learning Facies Classification Contest (Hall, 2016; Hall and Hall, 2017) and the TGS Salt Identification Challenge (Kaggle, 2019).

CNNs have dominated computer vision research in recent years, with deep CNNs becoming one of the most widely adopted methods for image classification across various domains (Nguyen et al., 2018). With an increase in depth of a neural network, performance is indeed improved, but this also results in an increase in computational resources and training time. Training CNNs from the ground up also typically require datasets containing millions of images across thousands of classes, such as ImageNet (<http://www.image-net.org/>; Deng et al., 2009).

Although many geoscience applications may involve large amounts of data, a common problem is the paucity of datasets with ground-truth labels. This is, in part, due to the laborious and expensive nature of geological data collection (Klump and Robertson, 2015). Datasets may also contain noise and missing values; while many variables cannot be measured directly, but only inferred from observations. Many geological applications rely on an iterative process of collecting different samples and subjectively applying descriptive geological interpretations to known instances. While data availability in geosciences is increasing, it is still comparatively small and sparse given the complexity of the phenomena under study. In deep learning, the limited representative training samples may result in poor performance of algorithms, for example due to overfitting, where the model is overly complex relative to the small dataset.

One promising method of applying deep CNNs to limited datasets and reducing training time is transfer learning (e.g. Pan and Yang, 2010; Shin et al., 2016; Tan et al., 2018), which utilises a pre-trained neural network to act as a feature extractor on a smaller dataset. The extracted features from a CNN structure, which has been trained on a significantly large image dataset, are thought to be generic and, therefore, applicable for learning on other image datasets (Yosinski et al., 2014). Several publications have successfully applied deep learning and transfer learning approaches to image-based lithological classifications of well logs, borehole images, thin sections, microtomographic (micro-CT) images, and core images (Table 1). Based on the results from these previous studies and owing to the relatively small size of geological datasets, transfer learning was identified as the best method for the image classification tasks in this work. This study will demonstrate how deep learning algorithms can be used to elicit information from digital carbonate core images to improve the reliability and objectivity of classifications. It will also demonstrate that despite the clear benefit of transfer learning, dataset size plays an important role and many geological datasets are still considered too small to train an algorithm that generalises well to unseen data.

## 2. Methodology

### 2.1. Computing environment

All training and evaluation stages were completed using Python version 3.8 (<https://www.python.org>) with the TensorFlow (version 2.0) (Abadi et al., 2016), tensorflow-keras backend (version 2.4) (Chollet, 2015), FastAI (Howard and Gugger, 2020), Scikit-learn (version 0.22) (Pedregosa et al., 2011), NumPy (version 1.17) (Oliphant, 2006), OpenCV (version 4.1.2) (Bradski, 2000) and Matplotlib (version 2.2.4) (Hunter, 2007) libraries. The deep learning frameworks were implemented using the CPU of a MacBook Pro 2019 model with a 2.4 GHz 8-Core Intel Core i9 processor and 64 GB of RAM, running the macOS Catalina 10.15.7 operating system.

### 2.2. Data preparation

The dataset for this study has been created using high-resolution core images scanned to TIFF format from the upper to distal carbonate slope transects drilled during ODP Leg 133 (Davies et al., 1991) and ODP Leg 194 (Isern et al., 2002) in the carbonate platforms and troughs of north-eastern Australia, and the platform carbonates drilled during IODP Expedition 359 in the Maldives archipelago (Betzler et al., 2017). High-resolution imaging of core sections has been routine since ODP Leg 198, and the digital core scanners used during ocean drilling expeditions currently capture approximately 10–20 pixels per millimetre of core (Wilkens et al., 2009). For the cores drilled during ODP Legs 133 and 194, the scans were performed by C. John in 2007 using a prototype line scanner at the Gulf Coast Repository, which is now onboard the JOIDES Resolution.

The information from the digital core images is represented by a 3D tensor with separate red, green, and blue (RGB) channels. Digital core scan images of a 1.5 m section are approximately 31,600 pixels x 1750 pixels; however, only the sediment core section itself is required here, which is on the order of 29,900 pixels x 1375 pixels within the image. These individual images can provide some of the highest resolution examples of core properties available at present.

#### 2.2.1. Image resizing

Many deep learning algorithms, including CNNs, require all images in a dataset to be resized to a unified dimension. For the dataset to be compatible with the original dimensions of the pre-trained architectures, and to leverage the natural-image features learned by the pre-trained networks, the core images used in this work were cropped using the slicing tool in Adobe Photoshop 2020 (version 21) (Adobe

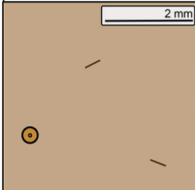




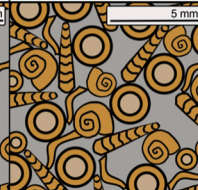
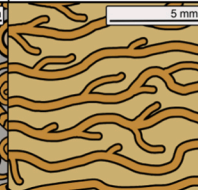
**Table 1**

Summary of techniques and comparison between studies for lithological classification.

Author(s) and Year	Dataset	Number of Classes	Dataset Size	Method	Model/Architecture	Result
Dubois et al. (2007)	Carbonate-Siliciclastic Well Logs	8	3647	Machine Learning	Custom ANN	78% (Accuracy)
Hall (2016)	Carbonate-Siliciclastic Well Logs	9	3232	Machine Learning	SVM	43% (F1-Score)
Tschannen et al. (2017)	Carbonate-Siliciclastic Well Logs	9	800	Transfer Learning(?)	GoogLeNet	57.40% (F1-Score)
Zhang et al. (2017)	Mixed Lithology Borehole Images	3	1500	Deep Learning	Custom CNN	95% (Accuracy)
Jobe et al. (2018)	Carbonate Thin Sections	6	~9000	Transfer Learning and Fine-Tuning	Inception-v3	83% (Accuracy)
Ivchenko et al. (2018)	Mixed Lithology Cores	4	800	Deep Learning	Custom CNN	88.7% (Accuracy)
Imamverdiyev and Sukhostat (2019)	Mixed Lithology Well Logs	9	4149	Deep Learning	Custom 1D-CNN	93.2% (Accuracy)
John and Kanagandran (2019)	Carbonate Cores	7	35,197	Transfer Learning	Inception-v3	89.2% (Accuracy)
Pires de Lima et al. (2019)	Carbonate-Siliciclastic Cores	11	7039	Transfer Learning	ResNetV2	95% (Accuracy)
Baraboshkin et al. (2020)	Mixed Lithology Cores	6	85,600	Transfer Learning	GoogLeNet	57% (F1-Score)
Pires de Lima et al. (2020)	Siltstone Thin Sections	5	5515	Transfer Learning and Fine-Tuning	ResNet50	96% (Accuracy)
Koeshidayatullah et al. (2020)	Carbonate Thin Sections	6	~13,000	Transfer Learning and Fine-Tuning	Inception-ResNet v2	92% (F1-Score)
Jiang et al. (2021)	Borehole-Resistivity Images	2	5156 (cleaned dataset)	Deep Learning	Custom CNN	84.7% (Accuracy)
dos Anjos et al. (2021)	Carbonate Rock Micro-CT Images	3	6000	Deep Learning	Custom CNN	81.33% (Accuracy)
Chawshin et al. (2021)	Mixed Lithofacies Core CT Scans	20	225,524	Deep Learning	Custom CNN	51.60% (F1-Score)

Inc., 2020) to  $224 \times 224$  pixels or  $299 \times 299$  pixels, according to the network input size. Downscaling images can cause loss of resolution, so each core image was sliced into multiple smaller images to preserve the original features of the carbonate textures. Digital core scan images of a 1.5 m section are approximately 31,600 pixels x 1750 pixels; however, only about 29,900 pixels x 1375 pixels within this image represent the core itself. A maximum of 400 images can, therefore, be produced from each whole core scan, dependent on recovery. Following the core

descriptions provided by the original expedition scientists as a base for the classifications, each core was analysed for the present classes, any boundaries were marked on the core and then the scan was sliced to avoid sampling multiple textures within a single image. To ensure the networks were extracting and learning features from the cores as opposed to other articles, such as Styrofoam inserts and core liners, all images were manually inspected and any images containing less than 70% core after slicing were removed from the dataset. This created a

Allochthonous						Autochthonous
Original components not bound together organically during deposition						Original components bound organically at deposition
Less than 10% components $\geq 2$ mm				More than 10% components $\geq 2$ mm		
Mud-supported		Grain-supported		Supported by matrix of components $< 2$ mm	Supported by components $\geq 2$ mm	
Less than 10% grains	More than 10% grains	Contains mud	Contains no mud			
Mudstone	Wackestone	Packstone	Grainstone	Floatstone	Rudstone	Boundstone
						

**Fig. 1.** The modified Dunham classification scheme showing the seven classes used in this research (after Dunham, 1962; Embry and Klovan, 1971).



dataset containing 104,306 images across seven classes from the modified Dunham Classification (Dunham, 1962; Embry and Klován, 1971): mudstone, packstone, wackestone, grainstone, floatstone, rudstone, and boundstone (Figs. 1 and 2).

To the best of our knowledge, this is the largest carbonate core image dataset used for image classification to date. In geoscience applications, however, there are many instances where only limited training data may be available. Therefore, in this paper, three datasets with orders of magnitude difference in data quantity were created, to allow further comparison of model performance across datasets of different sizes: the original 104k dataset, a 42k dataset, and a 7k dataset (Fig. 3).

### 2.2.2. Dataset splits

Prediction reliability is one of the main concerns in the performance evaluation of supervised deep learning algorithms (Consonni et al., 2010; Alsina et al., 2017). In this study, each dataset was split into 80% training data and 20% test data. We further used the training data with cross validation to create 5 folds of 80% training data and 20% validation data (i.e., validation set = 16% of the total dataset, Fig. 4). Since both the 48k and 104k datasets have a significant class imbalance, we used stratified k-fold cross-validation ( $k = 5$ ), where the percentage of samples for each class is maintained in every fold, ensuring the class distribution in each fold matched the distribution in the complete training dataset.

### 2.3. Model training

In our application, we use the learned parameters from the CNN architectures trained on the ImageNet dataset to classify core images according to the modified Dunham Classification, where classes are defined based on textural observations (Fig. 1; Dunham, 1962; Embry and Klován, 1971). Nine different convolutional neural networks (CNNs) of four different architectures were trained on the datasets (Table 2). The architectures selected for this comparison were DenseNet (Huang et al., 2017), Inception-v3 (Szegedy et al., 2016), ResNet (He et al., 2016) and VGG (Simonyan and Zisserman, 2014). Full details of the architectures are provided in their original references.

The pre-trained weights and parameters of the base convolutional networks made it possible to perform generic feature extraction from the

core images. The final layer of the network that performs classification was removed and replaced with a new classifier, which was trained on our carbonate core datasets (Fig. 5). The new classification head was implemented with the following architecture: an average pooling layer; a fully connected layer with 1024 hidden units and ReLU activation; a dropout rate of 0.2; and a final fully connected sigmoid layer. Following the fine-tuning method, each CNN was trained for a total of 25 epochs, whereby during the initial 15 epochs only the classification heads of the networks were trained. The uppermost layers of the networks were then unfrozen, and the models were trained as a whole for a further 10 epochs. This allowed us to fine-tune the learned higher-order feature representations to adapt their relevance for the specific core classification task.

Model training is a process by which differences between the predicted labels and the true labels of the training dataset are minimised through adjustments to weights and biases within the kernels of the convolutional layers and the fully connected layers (Yamashita et al., 2018). Network performance is evaluated during training using a cost function, which calculates a distance between the output predictions and the true label through forward-propagation. Since this is a multi-class classification, cross-entropy loss was used as our cost function, and the Softmax function was applied to the output of cross-entropy in order to derive a class for each sample. Learnable parameters are updated iteratively according to the loss value through backpropagation and gradient descent optimization algorithms, so as to minimise the loss. In this study, we evaluated the performance of two commonly used gradient descent algorithms: stochastic gradient descent (SGD) with momentum (Qian, 1999) and Adam (Kingma and Ba, 2014).

Using the one-cycle policy to improve training, reduce overfitting and allow the network to converge faster (Smith, 2017, 2018), the optimal learning rate was determined to be between  $1e - 2$  and  $1e - 3$  for the first 15 epochs and between  $1e - 5$  and  $1e - 6$  for the rest of the training. The one-cycle policy gives particularly fast results to train complex models, using a linear warm-up and annealing for the learning rate between the specified minimum and maximum learning rate boundaries. Batch size was held constant at 32 for all models. Training was repeated for a total of five times and predictions were averaged, since CNN performance can exhibit random minor fluctuations.

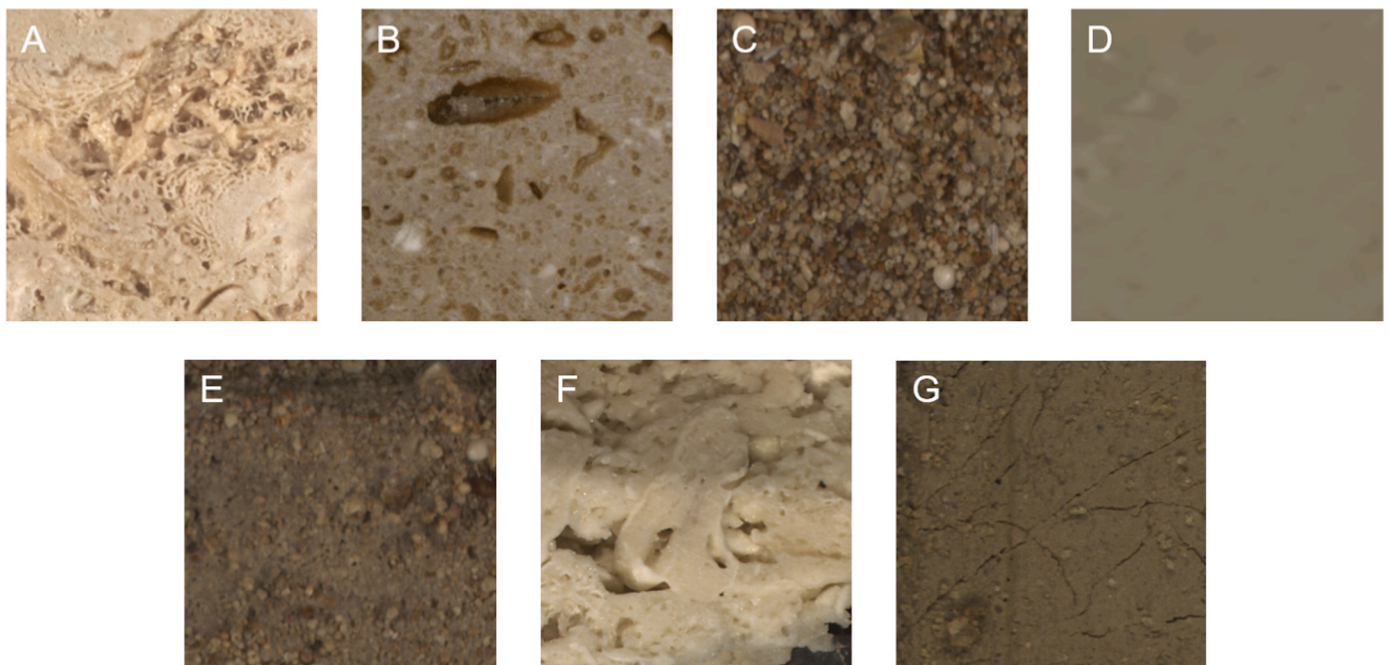


Fig. 2. Sample dataset images: (a) boundstone, (b) floatstone, (c) grainstone, (d) mudstone, (e) packstone, (f) rudstone, (g) wackestone.

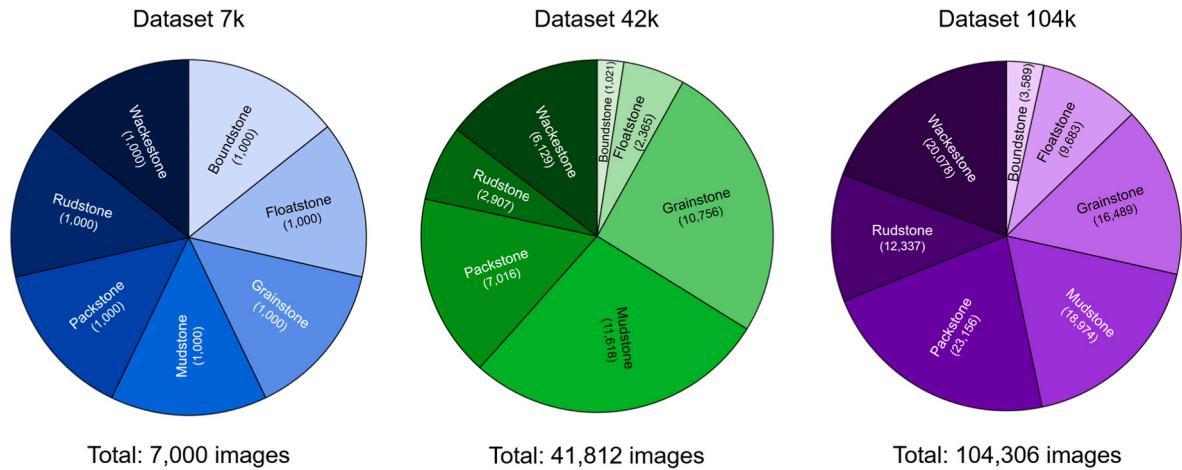


Fig. 3. Distribution of core images for each class across the three datasets.

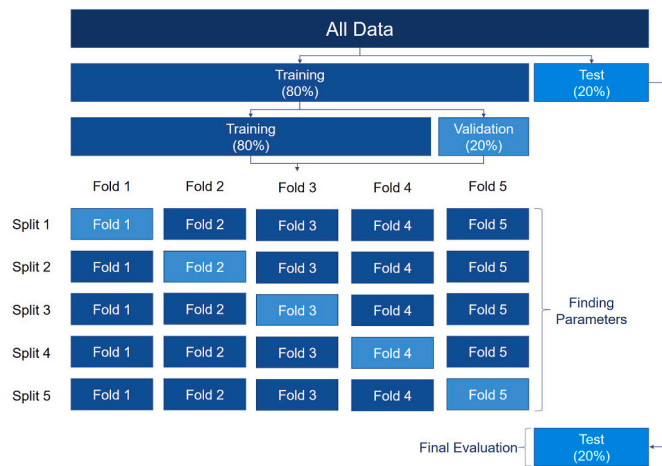


Fig. 4. Stratified k-fold cross validation ( $k = 5$ ) used in this study, where each dataset is divided into training, validation and test data.

Table 2

Summary of models used in this comparison.

Model	Size	Parameters	Depth	Image Input Size
DenseNet121	33 MB	8,062,504	121	224 × 224
DenseNet169	57 MB	14,307,880	169	224 × 224
DenseNet201	80 MB	20,242,984	201	224 × 224
Inception-v3	92 MB	23,851,784	48	299 × 299
ResNet50	98 MB	25,636,712	50	224 × 224
ResNet101	171 MB	44,707,176	101	224 × 224
ResNet152	232 MB	60,419,944	152	224 × 224
VGG16	528 MB	138,357,544	16	224 × 224
VGG19	549 MB	143,667,240	19	224 × 224

## 2.4. Evaluation

During training, the performance of the networks was evaluated by monitoring the changes in training and validation losses with each epoch. Following the completion of training, evaluation metrics (accuracy, precision and recall) and the confusion matrices (e.g. Ruuska et al., 2018; Skansi, 2018) were used to compare the prediction performances of the different CNNs on the image classes of the carbonate core test dataset. Accuracy measures the ratio between the number of correctly predicted classifications and the total number of samples classified:

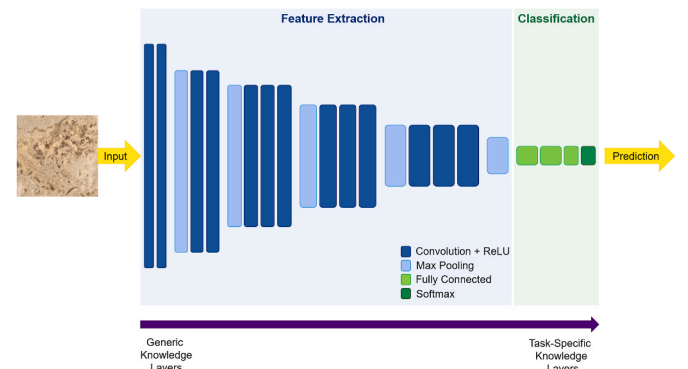


Fig. 5. Schematic representation of the CNN parts used in transfer learning (based on the VGG16 architecture). The feature extraction part of a CNN (shown in blue) is composed of convolutional and pooling layers and is used as the convolutional base. The main purpose of the convolutional base is to generate features from images, such as edges, lines and curves in the lower layers and shapes in the upper layers. The classification part of a CNN (shown in green) is usually composed of fully connected and Softmax layers. The classifier predicts the class of the input image based on task-specific features. (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

$$\text{Overall Accuracy : } ACC = \frac{TN + TP}{TN + TP + FN + FP} \quad (1)$$

whereby TN is the number of true negative cases, TP is the number of true positive cases, FN is the number of false negatives, and FP is the number of false positives. Since overall accuracy makes no distinction between classes, with correctly predicted classifications for each class being treated equally, this could be viewed as a misleading metric. This is particularly true in the case of significant class imbalances, where classes containing more samples will dominate the statistic. For this reason, we use multiclass averaging to report the mean per class accuracy (MPCA), given as the unweighted mean of the sum of the accuracies for each independent class:

$$MPCA = \frac{\sum ACC \text{ of Each Class}}{\text{Number of Classes}} \quad (2)$$

Precision calculates the proportion of positive identifications that were correct i.e., the probability that the predicted class of the carbonate facies actually belongs to that class:

$$\text{Precision} : P = \frac{TP}{TP + FP} \quad (3)$$

Recall calculates the proportion of actual positives that were correctly identified:

$$\text{Recall} : R = \frac{TP}{TP + FN} \quad (4)$$

We also consider how computational cost impacts classification performance by considering the model complexity, computational complexity, training time and classification speed of each architecture. By analysing the relationships between these metrics, we can further develop our understanding of the best practices for applying these models to geological images. Following Becker et al. (2021), model complexity is defined as the number of trainable parameters of the neural network. Computational complexity is defined in terms of floating-point operations (FLOPs) required by the model for a single forward pass.

### 3. Results

In most cases, for Dataset 42k and Dataset 104k, the best results were achieved using the Adam optimizer. For Dataset 7k, the SGD with momentum produced markedly better results for all but one of the models (Table 3).

#### 3.1. Performance evaluation for overall accuracy

On the large dataset (Dataset 104k), within the individual architectures, the deeper CNNs generally achieved a higher MPCA than shallower networks (Table 3, Fig. 6A). When considering the MPCA against the computational complexity of the architectures, within the VGG and DenseNet architectures, we clearly see an increase in MPCA with increasing depth and complexity. However, increased computational complexity does not always translate directly into a proportional increase in classification performance. A weak negative correlation between the MPCA and the FLOPs is present with an R2 value of 0.635. This is corroborated by the highest accuracy (91.59%) being obtained by Inception-v3, which shows comparatively lower model and computational complexities (Fig. 6A). The results observed for the larger dataset are similar to those for Dataset 42k (Fig. 6C), where Inception-v3 again achieves the highest accuracy (89.61%).

For the small dataset (Dataset 7k), we see a clear increase in MPCA with increasing computational complexity for a single forward pass (Fig. 6E). This is evidenced by the highest accuracy (85.11%) being achieved by VGG19, the most computationally complex architecture evaluated (20 GFLOPs). Those architectures with lower complexity achieve a lower MPCA e.g., DenseNet201 requires only 4 GFLOPs, but obtains the lowest MPCA (66.70%).

As is expected, within the individual architectures, we see the larger

and more complex networks requiring a greater training time (Fig. 6). For Dataset 7k, there is clear trade-off existing between faster training and higher recognition performance, with VGG19 achieving 85.11% accuracy with the longest training time of 219 min. By contrast, ResNet50 produces the second lowest accuracy of 70.16% after the shortest training time of 56 min. Despite the trend seen in the smaller dataset, for Datasets 42k and 104k, the highest MPCAs are achieved by Inception-v3, 89.61% and 91.59% respectively, whilst requiring a middle-of-the-range training time, 146 min and 194 min respectively. After this point, there is a general decrease in MPCA, suggesting these models may be training for too long and, as a result, are starting to overfit the training data.

#### 3.2. Performance evaluation for each class

We observe the top performing classifier (Inception-v3, Dataset 104k) predicts all of the classes (Dunham facies) with recall values above 0.86. The maximum precision observed was in the grainstone class (0.96) for DenseNet201 trained on Dataset 104k, and the best recall was in the mudstone class (0.97) for Inception-v3 trained on Dataset 104k (Figs. 7 and 8). Across all architectures and datasets, the lowest precision is observed in the floatstone class (0.73) (Fig. 7). Similarly, rudstones have the lowest recall (0.74) across all architectures (Fig. 8). The highest precision and recall values are achieved in five out of the seven classes by Inception-v3. It should be noted, that for the small dataset, since each class has only 1000 images, misclassifications are likely to have more of an impact on measured performance than those occurring in Dataset 42k and 104k.

### 4. Discussion

#### 4.1. Comparison with human performance and best model selection

The most successful model (Inception-v3, Dataset 104k) achieved an accuracy of nearly 92% (Table 3), also outperforming the other architectures in terms of precision, recall and computing time and resources. This significantly exceeds results achieved in previous carbonate classification studies, namely the 79% accuracy of carbonates facies prediction using random forest from physical properties (Insua et al., 2014) and the 89% accuracy achieved in the pilot neural network study (John and Kanagandran, 2019). This indicates that deep CNNs can provide a significant improvement on existing classifications, achieving a similar or better accuracy than experienced geologists (Lokier and Al Junaibi, 2016) with at least a 250x gain in description speed (Fig. 9).

The average classification speed of a person on the ImageNet dataset is estimated to be around 50 images per minute, which equates to 0.83 images per second (Markoff, 2012). It should be noted that the classification task from which this estimate was achieved, whilst relatively complex due to the large number of possible categories, is based on non-specialist images, and as such this estimate should be considered an upper bounding limit of the speed a person might be able to classify geological images. In comparison, an automated classification system based on the top performing model (Inception-v3, Dataset 104k) has a classification speed of 1179 images per second, calculated by inverting the inference time in which the test set (20,860 images) was classified. Even the model with the slowest classification speed on this dataset (VGG19) is able to classify 256 images per second (Fig. 9). In this manner, application of an automated system would allow for many more metres of core to be interpreted with lower human input and effort. As such, the role of a specialist geologist could be reallocated from lower-level descriptive tasks to QA/QC of automated classification, interpreting more complex images or exploring the broader meaning and implications of the data.

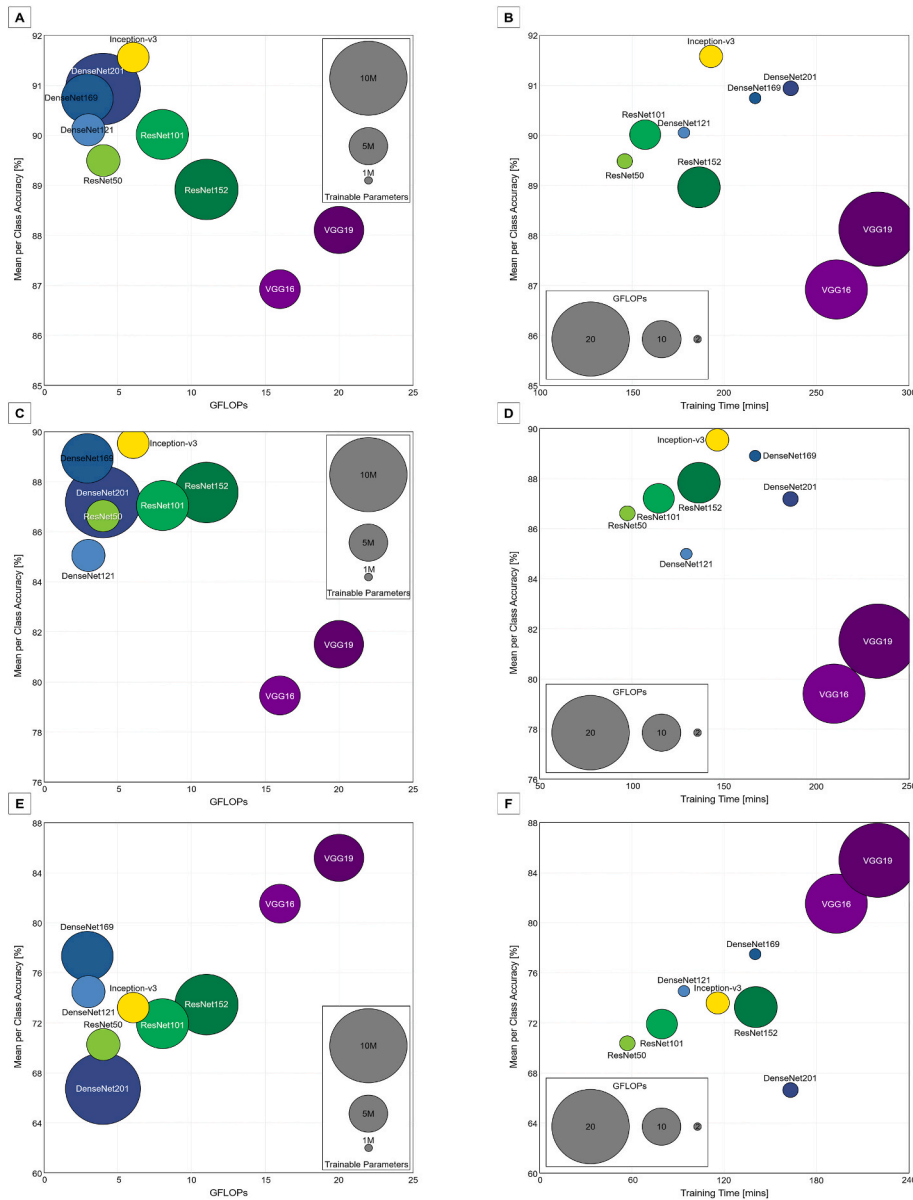
The use of CNNs with either fewer layers or lower computational complexity draws advantages in shorter training times and lesser

**Table 3**

Performance evaluation of classification accuracy during the prediction stage. The results for both optimizers under each dataset are shown for comparison.

Model	Accuracy					
	Dataset 7k		Dataset 42k		Dataset 104k	
	SGDM	Adam	SGDM	Adam	SGDM	Adam
DenseNet121	74.48	65.64	83.24	85.16	89.29	90.1
DenseNet169	77.37	67.12	84.47	88.94	<b>91.22</b>	90.76
DenseNet201	66.7	67.23	82.06	87.77	90.74	90.91
Inception-v3	73.12	65.99	<b>89.13</b>	<b>89.61</b>	91.14	<b>91.59</b>
ResNet50	70.16	61.73	86.05	86.78	88.46	89.49
ResNet101	71.92	63.18	86.53	87.09	89.07	90.03
ResNet152	73.44	65.52	87.79	87.73	89.62	88.91
VGG16	81.56	70.06	78.36	79.62	85.38	86.95
VGG19	<b>85.11</b>	<b>71.31</b>	81.17	81.63	87.93	88.26





**Fig. 6.** Performance evaluation of the CNN architectures on the three datasets. (A) Average accuracy vs computational complexity vs model complexity for Dataset 104k. (B) Average accuracy vs training time vs computational complexity for Dataset 104k. (C) Average accuracy vs computational complexity vs model complexity for Dataset 42k. (D) Average accuracy vs training time vs computational complexity for Dataset 42k. (E) Average accuracy vs computational complexity vs model complexity for Dataset 7k. (F) Average accuracy vs training time vs computational complexity for Dataset 7k.

hardware requirements compared to their deeper, more complex counterparts. The overall training process can be facilitated by these shorter training times as this could enable the integration and implementation of improvement methods such as “human in the loop” annotations; for example, where training is supervised by an expert geologist, who may assess any misclassifications showing high losses above a defined threshold in the validation set for incorrect labels, helping to reduce label noise. This could also allow for images with increased resolution, since there is potential for more images to be processed in a shorter training time, meaning larger images could be cropped into multiple smaller divisions rather than being downsampled. This could be of particular relevance for carbonate classification according to the Dunham classification, where specific divisions of the scheme are based on components measuring 2 mm or less; thus, requiring higher resolutions of input images, since crucial features could be lost due to downscaling.

As previously mentioned, the availability of large, labelled datasets is a major limitation for the application of deep learning in the geosciences. The comparisons in this paper are intended to aid anyone wanting to apply an image classification model to their own geological

dataset. Therefore, we draw recommendations for each dataset of different magnitude, so the most suitable model may be found for each. Comparing the results for Datasets 42k and 104k, we recognise similar trends in accuracy, precision, recall and computational resources. We would, therefore, again recommend the Inception-v3 architecture for datasets of this magnitude. Considering the smaller dataset (Dataset 7k), a clear trade off exists between computational complexity and higher classification performance (Fig. 6F). Therefore, for a smaller magnitude dataset, despite requiring more training time and computing resources, we would recommend the use of VGG19 due to the significant difference in accuracy achieved by this model over the other architectures evaluated.

#### 4.2. Effect of dataset size on prediction performance

There is an increase in prediction performance with increasing magnitude of the dataset size (Table 3) suggesting that dataset size affects prediction accuracy, even in transfer learning with CNNs. This is an important finding, as the majority of current studies use deep learning on datasets with under 10,000 samples (Table 1). When trained on the

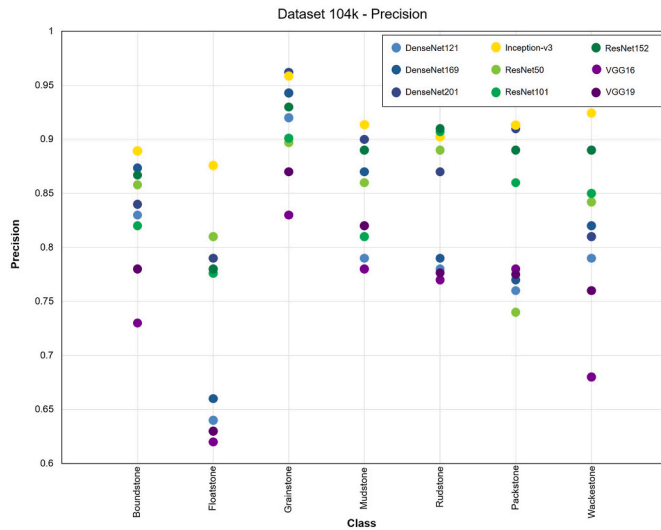


Fig. 7. Precision graph for Dataset 104k.

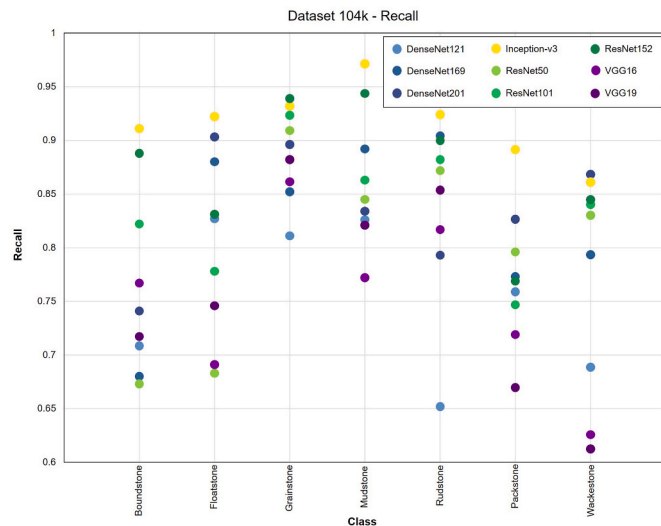


Fig. 8. Recall graph for Dataset 104k.

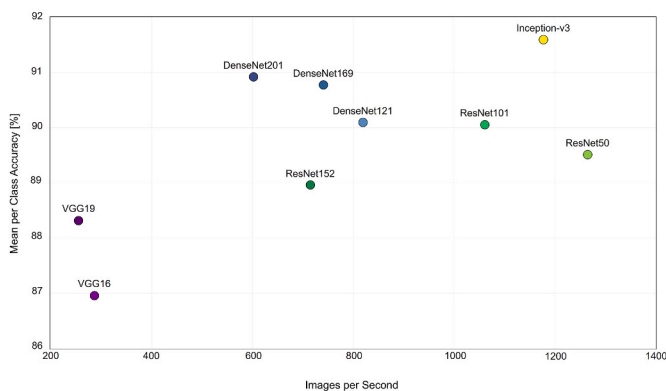


Fig. 9. MPCA vs images per second for architectures trained on Dataset 104k.

smaller dataset (Dataset 7k), although the Inception-v3 and ResNet architectures produce high training accuracies (>90%), much lower test accuracies are achieved (Table 3), suggesting likely overfitting, i.e. that the models are unable to generalise well to unseen data. Soekhoe et al.

(2016) found that for smaller target datasets, when using a fine-tuning approach, it is possible to reduce overfitting by leaving more layers frozen in the base model. By contrast, for large datasets, models can be improved through unfreezing and training more layers. Our results support this dichotomy, with the CNNs trained on Dataset 104k proving to be much more robust to overfitting.

Data augmentation is one way to artificially enlarge a training dataset, and reduce overfitting, by applying transformations to examples from the training data to create new images that belong to the same class as the original image. One of the most popular methods of data augmentation is the application of traditional affine and elastic transformations. This involves generating new images by applying procedures such as rotations, reflections, shifting, distortions, cropping, scaling, or colour shift. The tensorflow-Keras deep learning library in Python provides an easy way to incorporate augmentation using the ImageDataGenerator class (Keras Library); and the open-source Python library Albumentations offers a highly diverse set of more complex image transform operations that are optimised for different computer vision tasks, including image classification. Data augmentation techniques can, ultimately, help to expand a limited dataset, reduce overfitting, and improve the robustness of a model. However, it is important to ensure that the transformations applied to the images do not alter them in such a way that they are no longer recognisable by a carbonate geologist as, or a realistic representation of, carbonate rocks.

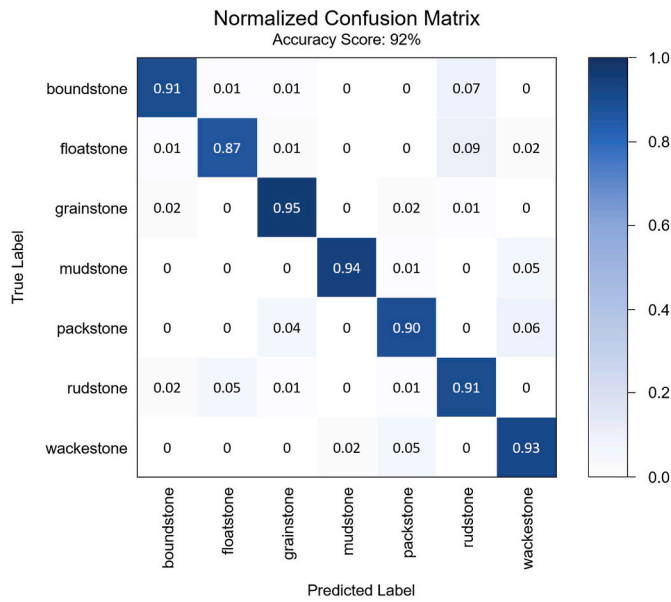
This study has highlighted that even when using transfer learning the size of the dataset plays a key role in the performance of the models, with those trained on the smaller datasets showing a strong tendency to overfit. Our results do, however, indicate that transferring features learned from a large source dataset to a much smaller target dataset still adds value by improving prediction performance and reducing risk of overfitting. It is shown that classification performance can continue to improve as more data is provided to the model; however, the capacity of the model needs to be regulated to support such data increases. Ultimately, there may be a point of diminishing returns where more data will not provide more insight into how to better improve classifications. For future applications applying transfer learning to geological images, we would recommend a minimum dataset size of 100,000 images which, in the case of limited samples, could be achieved through data augmentation techniques.

#### 4.3. Understanding sources of error and model biases

One of the major aims for this project is to reduce the subjectivity of carbonate facies classifications. Confusion matrix analysis highlights some specific errors affecting individual classes, which may be arising from sources of biases. We observe the main areas of misclassification are between the floatstone and rudstone classes, the boundstone and rudstone classes, the wackestone and packstone classes, and the mudstone and wackestone classes (Fig. 10). This suggests that the error is occurring between facies that are adjacent in the modified Dunham classification. Therefore, the trained networks errors are similar to errors a geologist could make. Furthermore, the results show that the CNN misclassified images of floatstone for rudstone and wackestone for packstone more frequently than the reverse, thus more frequently misclassifying matrix-supported textures as grain-supported textures. Despite this small saliency bias, the CNN performs more consistently and more accurately than the human classification results in the study by Lokier and Al Junaibi (2016).

The causes of this bias in the model are likely to stem from three possible sources. Firstly, the bias could have been introduced through the human labelling of the core images. Saliency is a known cognitive-psychological, ecological and evolutionary bias in humans (Korteling et al., 2018). As implemented in this study, when analysing a new image, a CNN will generate a set of probabilities that this image belongs to each of the learned microfacies. Since the current dataset consists solely of images classified by the lead author, it is plausible that an element of



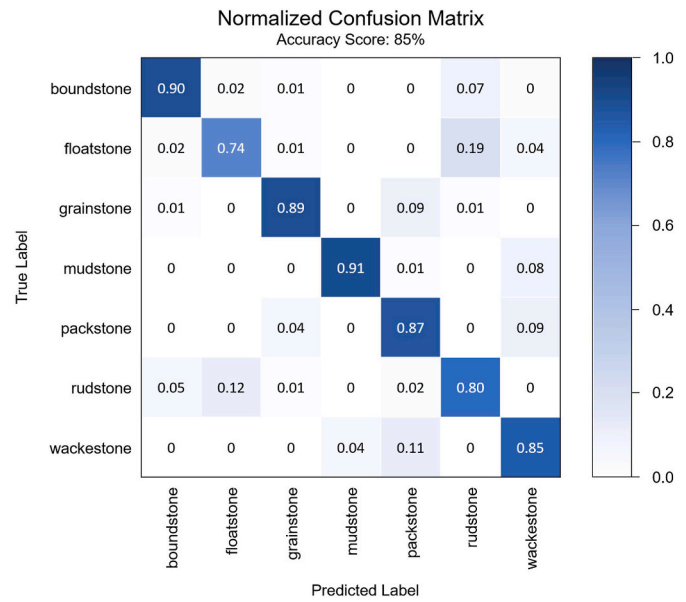


**Fig. 10.** Normalised confusion matrix for the Inception-v3 classifier on Dataset 104k. Average model accuracy = 92% for 7 facies.

bias could have been introduced into the algorithm training and, therefore, the final model. We acknowledge that this cannot be completely eliminated as a source of inaccuracy in this study. We aim to reduce interpreter bias in future work through the compilation of a dataset consisting of multiple classifications made by many individuals through a project currently being developed on an online crowdsourcing service. Citizen science already contributes to a variety of scientific fields, including astronomy, environmental science, ecology, medicine, and seismology (Theobald et al., 2015; Haklay et al., 2018). There are now thousands of projects collecting and processing data annually, with many contributing to peer-reviewed articles (Kelling et al., 2019; SciStarter.org, 2020). When properly designed, conducted, and evaluated, these projects can generate high-quality data to solve diverse problems.

Secondly, it is possible that the CNN architecture itself is biased towards salient features. Therefore, a cause of the error could be through image data encoding before classification. Where a geologist would rely upon a defined set of features and measurements to classify carbonate facies according to the modified Dunham scheme, the CNNs operate with no prior knowledge of specific attributes and, thus, perform the classifications based solely on image characteristics. Taking the misclassifications between mudstone and wackestone, for example, the feature vectors in an image of a mudstone would be sparse as mudstones tend to be relatively featureless. Hence, classification would be based on a sparse feature vector for this class. However, mudstones and wackestones are both mud-supported classes, with a subtle threshold difference at 10% grains. For borderline cases e.g., grainier mudstones containing 5–9% grains, the grains would become a significant feature and the CNN may place larger weights on these features, thus leading to a misclassification as a wackestone. To address this issue, introducing more labelled images of borderline cases to the dataset could likely further improve accuracy, as well as having multiple individuals classify these images to highlight any differences.

From this, it is logical to suggest that the bias may also be due to limited or imbalanced data. Considering the confusion matrix in Fig. 11, which is based on the balanced dataset, we observe that the misclassifications are again arising in the adjacent Dunham classes; floatstone and rudstone, boundstone and rudstone, wackestone and packstone, mudstone and wackestone, and, additionally, grainstone and packstone. To this effect, it should be noted that the balanced dataset is



**Fig. 11.** Normalised confusion matrix for the VGG19 classifier on Dataset 7k. Average model accuracy = 85% for 7 facies.

considerably smaller in magnitude, so we cannot rule out that the error here could, in part, be due to lower sampling.

## 5. Conclusions

This study builds on a new approach to carbonate core classification through deep learning, where image classification algorithms aim to improve overall interpretation accuracy, as well as reducing subjectivity and interpretation time. This paper evaluated the ability of nine different CNNs of four architectures to classify carbonate core images, systematically developing a performance comparison between these architectures on three complex geological datasets with orders of magnitude difference in data quantity. Following a transfer learning and fine-tuning approach, we used the learned parameters from the pre-trained CNN architectures to classify the core images according to the modified Dunham Classification, where classes are defined based on textural observations (Dunham, 1962; Embry and Klován, 1971).

The results show great potential for the use of deep learning in automated carbonate classification from digital core images, where high level performance was achieved across all models, even with limited and unbalanced datasets. The highest overall accuracy of 92% was achieved by the Inception-v3 architecture when trained on the larger (104k) dataset. When considering all evaluation metrics presented, for textural carbonate core classification, we find the Inception-v3 architecture to be the most suitable model for medium to large datasets, and the VGG19 architecture to be the most suitable for smaller datasets. Furthermore, we have shown that even when using transfer learning the size of the dataset plays a key role in the performance of the models, with those trained on the smaller datasets showing a strong tendency to overfit.

The use of machine learning in geoscience is rapidly growing; however, the lack of large, labelled training datasets presents a pivotal challenge in improving the prediction performance of CNNs for geological applications. As data availability in geosciences increases, the deep-learning approach we present here can be further improved with additional information generated from more recent data and the digitalisation of existing datasets from long-established sources. Ultimately, the development and application of the framework laid out in this study could aid the future of deep learning based carbonate classification and can easily be modified for the classification of cores originating from different lithologies and formations.

## Authorship contribution statement

Dawson, H.L. developed the code, performed the analyses, wrote the paper; Dubrule, O.J. reviewed and edited the paper, and supervised the project; John, C.M. designed the project, reviewed and edited the paper, and supervised the project.

## Code availability section

Name of the code/library: Transfer Learning for Geological Images  
Contact: [h.dawson19@imperial.ac.uk](mailto:h.dawson19@imperial.ac.uk).

## Hardware requirements

None.

## Program language

Python.

## Software required

Python 3.8 distribution (available for Windows, Linux, and macOS).

## Program size

10 KB.

The source codes are available for downloading at the link: <https://github.com/carbonateresearch/dawson-facies-2022>.

## Declaration of competing interest

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

## Data availability

Data will be made available on request.

## Acknowledgements

This research used data provided by the International Ocean Discovery Program (IODP).

## References

- Adobe Inc, 2020. Adobe Photoshop 2020. Available from: <https://www.adobe.com/uk/products/photoshop.html>.
- Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., et al., 2016. Tensorflow: a system for large-scale machine learning. In: 12th {USENIX} Symposium on Operating Systems Design and Implementation (OSDI), pp. 265–283.
- Alsina, E.F., Chica, M., Trawinski, K., Regattieri, A., 2017. On the use of machine learning methods to predict component reliability from data-driven industrial case studies. *Int. J. Adv. Manuf. Technol.* 94, 2419–2433.
- Baldwin, J.L., Bateman, R.M., Wheatley, C.L., 1990. Application of a neural network to the problem of mineral identification from well logs. *Log. Anal.* 3, 279–293.
- Baraboshkin, E.E., Ismailova, L.S., Orlov, D.M., Zhukovskaya, E.A., Kalmykov, G.A., Khotylev, O.V., Baraboshkin, E.Y., Koroteev, D.A., 2020. Deep convolutions for in-depth automated rock typing. *Comput. Geosci.* 135, 104330.
- Becker, B., Vaccari, M., Prescott, M., Grobler, T., 2021. CNN architecture comparison for radio galaxy classification. *Mon. Not. Roy. Astron. Soc.* 503 (2), 1828–1846.
- Bergen, K.J., Johnson, A.P., de Hoop, M.V., Beroza, G.C., 2019. Machine learning for data-driven discovery in solid Earth geoscience. *Science* 363, 1299.
- Bestagini, P., Lipari, V., Tubaro, S., 2017. A Machine Learning Approach to Facies Classification Using Well Logs. SEG Technical Program Expanded Abstracts, pp. 2137–2142, 2017.
- Betzler, C., Eberli, G.P., Alvarez Zarikian, C.A., And the expedition 359 scientists (2017) Maldives monsoon and sea level. Proceedings of the International Ocean Discovery Program: College Station, TX (International Ocean Discovery Program). doi: 10.14379/iodp.proc.359.
- Bradski, G., 2000. The OpenCV library. Dr. Dobb's J. Softw. Tools Prof. Program. 120, 122–125.
- Brown, W.M., Gedeon, T.D., Groves, D.L., Barnes, R.G., 2000. Artificial neural networks: a new method for mineral prospectivity mapping. *Aust. J. Earth Sci.* 47 (4), 757–770. <https://doi.org/10.1046/j.1440-0952.2000.00807.x>.
- Chawshin, K., Berg, C.F., Varagnolo, D., Lopez, O., 2021. Lithology classification of whole core CT scans using convolutional neural networks. *SN Appl. Sci.* 3, 668.
- Chevitarese, D.S., Szwarcman, D., Brazil, E.V., Zadrozny, B., 2018. Efficient classification of seismic textures. In: International Joint Conference on Neural Networks (IJCNN), 2018, pp. 1–8. <https://doi.org/10.1109/IJCNN.2018.8489654>.
- Chollet, F., 2015. Keras. <https://github.com/fchollet/keras>.
- Consonni, V., Ballabio, D., Todeschini, R., 2010. Evaluation of model predictive ability by external validation techniques. *J. Chemometr.* 24, 194–201.
- Davies, P.J., McKenzie, J.A., Palmer-Julson, A., et al., 1991. Principal results and summary. In: Davies, P.J., McKenzie, J.A., Palmer-Julson, A., et al. (Eds.), Proceedings of the Ocean Drilling Program, 133, pp. 73–134. Initial Reports.
- de Matos, M.C., Osorio, P.L., Johann, P.R., 2007. Unsupervised seismic facies analysis using wavelet transform and self-organizing maps. *Geophysics* 72 (1), 9–21. <https://doi.org/10.1190/1.2392789>.
- Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L., 2009. ImageNet: a large-scale hierarchical image database. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255.
- dos Anjos, C.E., Avila, M.R., Vasconcelos, A.G., Neta, A.M.P., Medeiros, L.C., Evsukoff, A. G., Surmas, R., Landau, L., 2021. Deep learning for lithological classification of carbonate rock micro-CT images. *Comput. Geosci.* 25, 971–983.
- Dubois, M.K., Bohling, G.C., Chakrabarti, S., 2007. Comparison of four approaches to a rock facies classification problem. *Comput. Geosci.* 33 (5), 599–617.
- Dunham, R.J., 1962. Classification of carbonate rocks according to depositional texture. In: Ham, W.E. (Ed.), Classification of Carbonate Rocks, vol. 1. American Association of Petroleum Geologists Memoirs, pp. 108–121.
- Embry, A.F., Klován, J.E., 1971. A late devonian reef tract on northeastern banks island, NWT. *Bull. Can. Petrol. Geol.* 19, 730–781.
- Haklay, M., Mazumdar, S., Wardlaw, J., 2018. Citizen science for observing and understanding the Earth. In: Mathieu, P.P., Aubrecht, C. (Eds.), Earth Observation Open Science and Innovation, vol. 15. ISSI Scientific Report Series, pp. 69–88. [https://doi.org/10.1007/978-3-319-65633-5\\_4](https://doi.org/10.1007/978-3-319-65633-5_4).
- Hall, B., 2016. Facies classification using machine learning. *Lead. Edge* 35 (10), 906–909.
- Hall, M., Hall, B., 2017. Distributed collaborative prediction: results of the machine learning contest. *Lead. Edge* 36 (3), 267–269.
- Haloteli, J., Demyanov, V., Gardiner, A., 2020. Value of geologically derived features in machine learning facies classification. *Math. Geosci.* 52, 5–29.
- He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition. CVPR, pp. 770–778. <https://doi.org/10.1109/CVPR.2016.90>, 2016.
- Howard, J., Gugger, S., 2020. Fastai: a layered API for deep learning. *Information* 11 (2), 108. <https://doi.org/10.3390/info11020108>.
- Huang, G., Liu, Z., Weinberger, K.Q., 2017. Densely Connected Convolutional Networks. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2261–2269.
- Hull, D., Chapman, P., Miller, D., Ingraham, D., Fritz, N., Kernan, N., 2015. Regional Eagle Ford Modeling: Integrating Facies, Rock Properties, and Stratigraphy to Understand Geologic and Reservoir Characteristics. Unconventional Resources Technology Conference, San Antonio, Texas. <https://doi.org/10.15530/urtec-2015-2173648>.
- Hunter, J.D., 2007. Matplotlib: a 2D graphics environment. *Comput. Sci. Eng.* 9 (3), 90–95.
- Imamverdiyev, Y., Sukhostat, L., 2019. Lithological facies classification using deep convolutional neural network. *J. Petrol. Sci. Eng.* 174, 216–228.
- Insua, T.L., Hamel, L., Moran, K., Anderson, L.M., Webster, J.M., 2014. Advanced classification of carbonate sediments based on physical properties. *Sedimentology* 62 (2), 590–606.
- Isern, A.R., Anselmetti, F.S., Blum, P., et al., 2002. Leg 194 summary. In: Proceedings of the Ocean Drilling Program, pp. 1–88. Initial Reports 194.
- Ivchenko, A.V., Baraboshkin, E.E., Ismailova, L.S., Orlov, D.M., Koroteev, D.A., Baraboshkin, E.Y., 2018. Core photo lithological interpretation based on computer analyses. In: IEEE Northwest Russia Conference on Mathematical Methods in Engineering and Technology, pp. 425–428.
- Jiang, J., Xu, R., James, S.C., Xu, C., 2021. Deep-learning-based vuggy facies identification from borehole images. *SPE Reservoir Eval. Eng.* 24, 250–261, 01.
- Jobe, T.D., Vital-Brazil, E., Khat, M., 2018. Geological feature prediction using image-based machine learning. *Petrophysics* 59 (6), 750–760.
- John, C.M., Kanagandran, S., 2019. AI to Improve the Reliability and Reproducibility of Descriptive Data: a Case Study Using Convolutional Neural Networks to Recognize Carbonate Facies in Cores. AAPG Annual Convention and Exhibition, San Antonio, TX.
- Kaggle, 2019. TGS Salt Identification Challenge. <https://www.kaggle.com/c/tgs-salt-identification-challenge>.
- Karpatne, A., Ebert-Uphoff, I., Ravela, S., Babaie, H.A., Kumar, V., 2019. Machine learning for the geosciences: challenges and opportunities. *IEEE Trans. Knowl. Data Eng.* 31 (8), 1544–1554. <https://doi.org/10.1109/TKDE.2018.2861006>.
- Kelling, S., Johnston, A., Bonn, A., Fink, D., Ruiz-Gutierrez, V., Bonney, R., Fernandez, M., Hochachka, W.M., Julliard, R., Kraemer, R., Guralnick, R., 2019. Using semistructured surveys to improve citizen science data for monitoring biodiversity. *Bioscience* 69 (3), 170–179.

- Kingma, D.P., Ba, J., 2014. Adam: A Method for Stochastic Optimization arXiv preprint arXiv:1412.6980.
- Klump, J.F., Robertson, J., 2015. Synthetic geology - exploring the "what if?". In: *Geology. AGU Fall Meeting Abstracts*, American Geophysical Union, Fall Meeting, pp. IN51A-1783, 2015.
- Koeshidayatullah, A., Morsilli, M., Lehrmann, D.J., Al-Ramadan, K., Payne, J.L., 2020. Fully automated carbonate petrography using deep convolutional neural networks. *Mar. Petrol. Geol.* 122, 104687.
- Korteling, J.E., Brouwer, A.-M., Toet, A., 2018. A neural network framework for cognitive bias. *Front. Psychol.* 9 (1561) <https://doi.org/10.3389/fpsyg.2018.01561>.
- Kortström, J., Uski, M., Tiira, T., 2016. Automatic classification of seismic events within a regional seismograph network. *Comput. Geosci.* 87, 22–30.
- Lokier, S.W., Al Junaibi, M., 2016. The petrographic description of carbonate facies: are we all speaking the same language? *Sedimentology* 63, 1843–1885.
- Markoff, J., 2012. Seeking a Better Way to Find Web Images. *New York Times*. <https://www.nytimes.com/2012/11/20/science/for-web-images-creating-new-technology-to-seek-and-find.html>. (Accessed 20 November 2012).
- Mosser, L., Dubrule, O., Blunt, M.J., 2020. Stochastic seismic waveform inversion using generative adversarial networks as a geological prior. *Math. Geosci.* 52, 53–79.
- Mousavi, S.M., Zhu, W., Sheng, Y., Beroza, G.C., 2019. CRED: a deep residual network of convolutional and recurrent units for earthquake signal detection. *Sci. Rep.* 9 <https://doi.org/10.1038/s41598-019-45748-1>, 10267.
- Nguyen, L.D., Lin, D., Lin, Z., Cao, J., 2018. Deep CNNs for microscopic image classification by exploiting transfer learning and feature concatenation, 2018. In: *IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1–5. Florence.
- Oikonomou, D., Alaei, B., Larsen, E., Jackson, C.A.-L., 2017. Machine Learning in Petroleum Geoscience: Constructing EarthNET. *NGF Winter Conference*, Oslo, Norway, , January 2017.
- Oliphant, T.E., 2006. A guide to NumPy. *Methods* 1, 85.
- Pan, S.J., Yang, Q., 2010. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* 22 (10), 1345–1359.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perot, M., Duchesnay, É., 2011. Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* 12, 2825–2830.
- Pires de Lima, R., Suriamin, F., Marfurt, K.J., Pranter, M.J., 2019. Convolutional neural networks as aid in core lithofacies classification. *Interpretation* 7, SF27–SF40.
- Pires de Lima, R., Duarte, D., Nicholson, C., Slatt, R., Marfurt, K.J., 2020. Petrographic microfacies classification with deep convolutional neural networks. *Comput. Geosci.* 142, 104481.
- Qian, N., 1999. On the momentum term in gradient descent learning algorithms. *Neural Network*. 12, 145–151.
- Qian, F., Yin, M., Liu, X.Y., Wang, Y.J., Lu, C., Hu, G.M., 2018. Unsupervised seismic facies analysis via deep convolutional autoencoders. *Geophysics* 83 (3), A39–A43.
- Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., Prabhat, 2019. Deep learning and process understanding for data-driven Earth system science. *Nature* 566, 195–204.
- Ruuska, S., Hämäläinen, W., Kajava, S., Mughal, M., Matilainen, P., Mononen, J., 2018. Evaluation of the confusion matrix method in the validation of an automated system for measuring feeding behaviour of cattle. *Behav. Process.* 148, 56–62. <https://doi.org/10.1016/j.beproc.2018.01.004>.
- Saporetto, C.M., da Fonseca, L.G., Pereira, E., de Oliveira, L.C., 2018. Machine learning approaches for petrographic classification of carbonate-siliciclastic rocks using well logs and textural information. *J. Appl. Geophys.* 155, 217–225.
- SciStarter.org, 2020. Project Finder. SciStarter. Available from: <https://scistarter.org/finder>.
- Shin, H.C., Roth, H.R., Gao, M., Lu, L., Xu, Z., Nogues, I., Yao, J., Mollura, D., Summers, R.M., 2016. Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. *IEEE Trans. Med. Imag.* 35 (5), 1285–1298.
- Shoji, D., Noguchi, R., Otsuki, S., Hino, H., 2018. Classification of volcanic ash particles using a convolutional neural network and probability. *Sci. Rep.* 8 (8111), 1–12.
- Simonyan, K., Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. In: *International Conference on Learning Representations*, pp. 1–14.
- Skansi, S., 2018. Introduction to Deep Learning: from Logical Calculus to Artificial Intelligence. Springer, Berlin, p. 191. <https://doi.org/10.1007/978-3-319-73004-2>.
- Smith, L.N., 2017. Cyclical learning rates for training neural networks. In: *IEEE Winter Conference on Applications of Computer Vision. WACV*, pp. 464–472. <https://doi.org/10.1109/WACV.2017.58>, 2017.
- Smith, L.N., 2018. A Disciplined Approach to Neural Network Hyper-Parameters: Part 1—learning Rate, Batch Size, Momentum, and Weight Decay arXiv preprint arXiv: 1803.09820.
- Soekhoe, D., van der Putten, P., Plaat, A., 2016. On the impact of data set size in transfer learning using deep neural networks. In: *Bostrom, H., Knobbe, A., Soares, C., Papapetrou, P. (Eds.), Advances in Intelligent Data Analysis XV. IDA 2016. Lecture Notes in Computer Science 9897*. Springer, Cham. [https://doi.org/10.1007/978-3-319-46349-0\\_5](https://doi.org/10.1007/978-3-319-46349-0_5).
- Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.B., 2016. Rethinking the inception architecture for computer vision. In: *IEEE Conference on Computer Vision and Pattern Recognition. CVPR*, pp. 2818–2826. <https://doi.org/10.1109/CVPR.2016.308>, 2016.
- Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., Liu, C., 2018. A survey on deep transfer learning. In: *Kürkova, V., Manolopoulos, Y., Hammer, B., Iliadis, L., Maglogiannis, I. (Eds.), Artificial Neural Networks and Machine Learning – ICANN 2018. ICANN 2018. Lecture Notes in Computer Science*, vol. 11141. Springer, Cham. [https://doi.org/10.1007/978-3-030-01424-7\\_27](https://doi.org/10.1007/978-3-030-01424-7_27).
- Theobald, E.J., Ettinger, A.K., Burgess, H.K., DeBey, L.B., Schmidt, N.R., Froehlich, H.E., Wagner, C., HilleRisLambers, J., Tewksbury, J., Harsch, M.A., Parrish, J.K., 2015. Global change and local solutions: tapping the unrealized potential of citizen science for biodiversity research. *Biol. Conserv.* 181, 236–244.
- Torrisi, F., Folzani, F., Corradino, C., Amato, E., Del Negro, C., 2021. Detecting volcanic ash plume components from space using machine learning techniques. *AGU 2021 Fall Meeting*. <https://doi.org/10.1002/essoar.10509947.1>.
- Tschannen, V., Delescluse, M., Rodriguez, M., Keuper, J., 2017. Facies classification from well logs using an inception convolutional network. arXiv preprint. 1706.00613, pp. 1–5.
- West, B.P., May, S.R., Eastwood, J.E., Rossen, C., 2002. Interactive seismic facies classification using textural attributes and neural networks. *Lead. Edge* 21 (10), 1042–1049.
- Wilkens, R.H., Niklis, N., Frazer, M., 2009. Data report: digital core images as data: an example from IODP Expedition 303. In: *Channell, J.E.T., Kanamatsu, T., Sato, T., Stein, R., Alvarez Zarikian, C.A., Malone, M.J., the (Eds.), Expedition 303/306 Scientists, Proceedings of the Integrated Ocean Drilling Program, 303/306*, pp. 1–16.
- Yamashita, R., Nishio, M., Do, R.K.G., Togashi, K., 2018. Convolutional neural networks: an overview and application in radiology. *Insight Image.* 9, 611–629. <https://doi.org/10.1007/s13244-018-0639-9>.
- Yosinski, J., Clune, J., Bengio, Y., Lipson, H., Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q., 2014. How transferable are features in deep neural networks?. In: *Proceedings of the 27th International Conference on Neural Information Processing Systems (NIPS'14)*, 2, pp. 3320–3328.
- Zhang, P.Y., Sun, J.M., Jiang, Y.J., Gao, J.S., 2017. Deep learning method for lithology identification from borehole images. In: *Paper Presented at the EAGE Conference and Exhibition, Paris, France, 12-15 June*.