Journal of Big Data

RESEARCH Open Access

Efficient pollen grain classification using pre-trained Convolutional Neural Networks: a comprehensive study

Masoud A. Rostami[1*], Behnaz Balmaki[2,3], Lee A. Dyer[4], Julie M. Allen[3], Mohamed F. Sallam[5] and Fabrizio Frontalini[6]

*Correspondence:
masoud.rostami@uta.edu

[1] Data Science Program, University of Texas at Arlington, Arlington, TX 76019, USA
[2] Department of Biology, University of Texas at Arlington, Arlington, TX 76019, USA
[3] Department of Biological Sciences, Virginia Tech, Blacksburg, VA 24060, USA
[4] Department of Biology, University of Nevada, Reno, Reno, NV 89503, USA
[5] Preventive Medicine and Biostatistics Department, Uniformed Service University of the Health Sciences, Bethesda, MD 20814, USA
[6] DiSPeA, University of Urbino Carlo Bo, Campus Scientifico Enrico Mattei, Località Crocicchia, 61029 Urbino, Italy

Abstract

Pollen identification is necessary for several subfields of geology, ecology, and evolutionary biology. However, the existing methods for pollen identification are laborious, time-consuming, and require highly skilled scientists. Therefore, there is a pressing need for an automated and accurate system for pollen identification, which can be beneficial for both basic research and applied issues such as identifying airborne allergens. In this study, we propose a deep learning (DL) approach to classify pollen grains in the Great Basin Desert, Nevada, USA. Our dataset consisted of 10,000 images of 40 pollen species. To mitigate the limitations imposed by the small volume of our training dataset, we conducted an in-depth comparative analysis of numerous pre-trained Convolutional Neural Network (CNN) architectures utilizing transfer learning methodologies. Simultaneously, we developed and incorporated an innovative CNN model, serving to augment our exploration and optimization of data modeling strategies. We applied different architectures of well-known pre-trained deep CNN models, including AlexNet, VGG-16, MobileNet-V2, ResNet (18, 34, and 50, 101), ResNeSt (50, 101), SE-ResNeXt, and Vision Transformer (ViT), to uncover the most promising modeling approach for the classification of pollen grains in the Great Basin. To evaluate the performance of the pre-trained deep CNN models, we measured accuracy, precision, F1-Score, and recall. Our results showed that the ResNeSt-110 model achieved the best performance, with an accuracy of 97.24%, precision of 97.89%, F1-Score of 96.86%, and recall of 97.13%. Our results also revealed that transfer learning models can deliver better and faster image classification results compared to traditional CNN models built from scratch. The proposed method can potentially benefit various fields that rely on efficient pollen identification. This study demonstrates that DL approaches can improve the accuracy and efficiency of pollen identification, and it provides a foundation for further research in the field.

Keywords: Pollen identification, Deep learning, Transfer learning, Convolutional Neural Networks, Great basin

Background and literature review

The identification and classification of pollen grains are essential methods for various fields, including agriculture, ecology, paleoclimatology, agriculture, environment, paleoecology, archeology, medicine, and forensics [1–4]. The field of pollen grain taxonomy, known as Palynology, relies heavily on analyzing morphological characteristics such as general shape, polarity, symmetry, apertures, size, and ornamentation. However, due to the frequent morphological similarities among pollen grains, it can be challenging to use these features to quickly and accurately identify pollen species, genera, or even families (as illustrated in Fig. 1), and traditional identification methods have been associated with high error rates [5–7]. Also, manually identifying pollen grains using microscopes is time-consuming and labor-intensive.

Automating the identification process using DL algorithms can provide several benefits, including reducing the time and effort required for identification, improving the accuracy and consistency of the results, and enabling large-scale analysis of pollen grain samples. These methods can lead to new insights and discoveries in numerous fields. In recent years, the demand for high accuracy and computational efficiency has increased in industry and academia due to the availability of advanced technology in computer vision and image processing. Deep learning has been widely utilized to maximize efficiency and accuracy, reduce labor, and minimize artifacts [8–12].

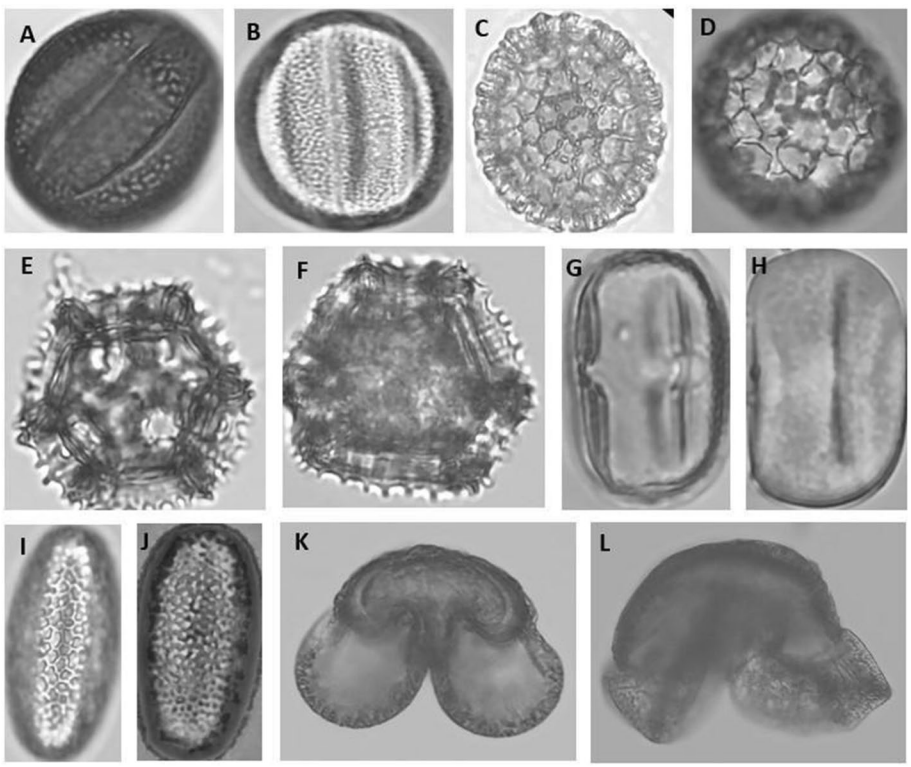

Fig. 1 Images of pollen grains representing the similarities in morphological features across divergent taxa. **A** *Salvia dorii*; **B** *Monardella villosa*; **C** *Phlox longifloia*; **D** *Phlox diffusa*; **E** *Taraxacum officinale*; **F** *Taraxacum californicum*; **G** *Astragalus pulsiferae*; **H** *Astragalus purshii*; **I** *Erysimum capitatum*; **J** *Sisymbrium altissimum*; **K** *Pinus monophylla*; **L** *Abies concolor*

Among the DL techniques, CNNs have gained popularity over the past decade for image classification, object detection, and task recognition, owing to their powerful neural network architecture that automatically extracts mid- to high-level features from image datasets [13–15].

CNN modeling has been proven effective for pollen taxonomic classification, especially when using transfer learning, which involves pre-trained CNN models to solve new problems [3–5, 7]. However, CNN models require massive training datasets, making pollen grain classification challenging due to the limited availability of pollen images. Transfer learning is an effective technique for learning features from small training datasets and automatically classifying images, making it a powerful tool for deep networking training without overfitting. One limitation, however, is that transfer learning heavily rely on large datasets to avoid overfitting [16–18]. Transfer learning is a technique in which a pre-trained CNN model, trained on a large dataset such as ImageNet that contains millions of images, is repurposed to learn a new task by leveraging the knowledge already gained from the previous task [19]. In the context of pollen classification, these pre-trained models can be used to make predictions or combined to train a new model. Transfer learning offers several advantages, including reducing the amount of time required to train a model from scratch, which is typically time-consuming and requires many parameter combinations. Moreover, utilizing pre-trained models can lead to higher accuracy and a lower risk of overfitting, making it a valuable approach for pollen classification tasks [17, 20, 21].

Previous studies on pollen grain automation have succeeded to some extent [4, 5, 7]. However, one of the main challenges in identifying pollen species is the limited availability of pollen datasets for training neural network models. The small number of datasets makes it difficult to define relevant features and variations in pollen morphology for identification purposes, especially given the similarities among pollen species [19, 22–29]. Moreover, most previous studies on pollen identification algorithms have primarily focused on Europe, Asia, and equatorial regions, leaving a gap in the literature for North America [23–29]. Our study is fills this gap by focusing on pollen classification in North America. Studying pollen grains in this region can expand our understanding of pollen morphology and provide more accurate identification tools for researchers worldwide.

The aim of this study was to enhance the accuracy of pollen grain image classification by utilizing transfer learning techniques to address the challenge of limited training data. To accomplish this objective, we employed a total of eleven transfer learning architectures, namely AlexNet, VGG-16, MobileNet-V2, ResNet (18, 34, 50, 101), ResNeSt (50, 101), SE-ResNeXt, and ViT, in addition to developing a CNN model from scratch. Our research objective was to achieve highly accurate and efficient classification of pollen grain images in North America, which has not been previously accomplished. Furthermore, this study sought to answer several critical scientific questions, such as the effectiveness of the proposed scratch CNN model in identifying pollen grains compared to the transfer learning models, and how the performance of the 11 transfer learning models compared to each other in identifying different types of pollen grains. We also aimed to investigate the limitations of the proposed models in identifying pollen grains and provide possible avenues for addressing these limitations in future research.

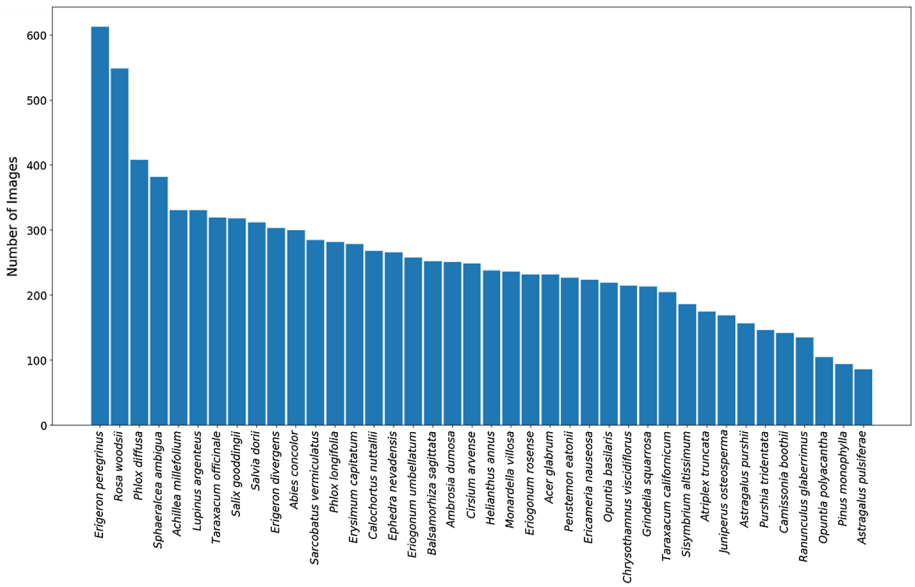

Fig. 2 Historgram of taxa images used in this study

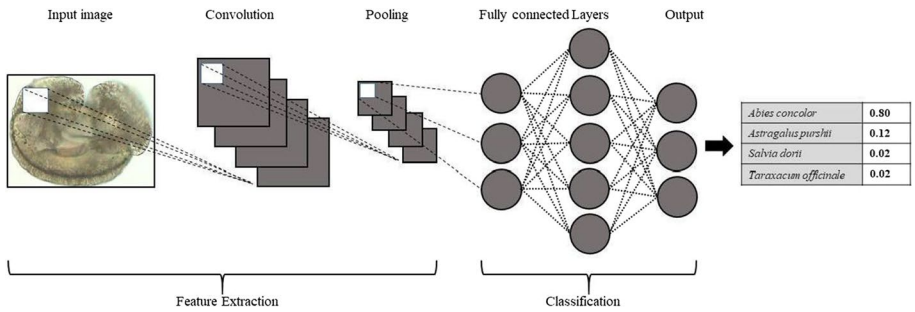

Fig. 3 The basic structure of CNN

Materials and methods

Data collection and data preprocessing

To train and test the model, we needed to collect a dataset of images of pollen grains. In this study, pollen grains were collected from plants located at the University of Nevada, Reno Museum of Natural History (UNRMNH). To ensure the accuracy of pollen identifications, the researchers prepared over 400 reference slides containing pollen from previously identified native flowers at the UNRMNH. A total of 10,000 images from the 400 pollen reference slides, representing 40 pollen species, were taken and used for training the model, where each class includes a range between 95 and 500 (Figs. 2, 3) images of size 224×224 and in *.jpg format.

We used a ZEISS Axiolab 5 light microscope and an Axiocam 208 color microscope camera to identify and photograph pollen grains. The images were captured using $40\times$ objective lenses and $10\times$ ocular lenses. Z-stack images were used to capture all details of the pollen grains, showing the vertical details of pollen grains at various focus levels. To prepare the images for training the model, we cropped each image

using Adobe Photoshop (CS6, 13.0.1.3). Then, we removed images with high noise levels due to debris, air bubbles, or aggregated pollen.

Before training the models, the dataset was preprocessed; this step includes normalizing the pixel values to a specific range and resizing the images to the appropriate input size for the models. We also applied data augmentation techniques such as rotating, flipping, or adding noise to the images to improve model robustness and prevent overfitting. The dataset was split into training (70%), validation (15%), and test sets (15%) to train the models.

CNN modeling background

CNNs are a type of DL model used for image classification tasks. These models comprise multiple layers, including input, hidden, and output layers (Fig. 3). The input layer takes the image dataset as input, which is then preprocessed and resized to an optimal size and passed to the convolutional layer. In the convolutional layer, filters or kernels perform element-wise multiplications with input images to extract low and high-level features, while the pooling layer reduces the size of the image while retaining important information. Next, normalization (ReLU) is applied to the features extracted in the convolutional layer, followed by processing in the fully connected layers, where the images are processed with a non-linear function to produce distinct categories with probabilities ranging from 0 to 1 for each taxon. This step adds considerable power to traditional taxonomic approaches, while the automated classification step provides a quick computerized approach for identifying pollen. The output layer provides the final classification result for the given input image.

Research methodology

Create a model from scratch

We developed the CNN model with a 6-layer model created from scratch. We chose an input image size of 224×224 and applied data augmentation techniques like rotation, rescale, shear, zoom, and horizontal flip to the training image data. The Rectified Linear Unit (ReLU) activation function was used within each convolutional layer. To avoid overfitting, a dropout with a rate of 0.2 was implemented. The softmax function was applied to estimate the probability for each taxon. The model consisted of three convolutional layers and two fully connected layers. The Adam optimizer with a learning rate of 0.0001 was used for training and trained the model for 14 epochs with a batch size of 32.

Transfer learning

Transfer learning was utilized as a technique to improve the classification accuracy of pollen grain images. The approach involves using a pre-trained CNN model as the starting point for a new task. The weights and biases of some layers are unfrozen and trained on the new image dataset, allowing the pre-trained model to adapt to the new task. The model architecture is adjusted by freezing some layers of the pre-trained model and modifying the output neurons to fit the specific needs of the task. The convolutional layers act as a fixed feature extractor that extracts relevant features from the input pollen images for classification. For retraining these transferred networks, the number of

classes in the last layer was adjusted to 40, which is the number of pollen species present in the Great Basin.

Proposed transfer learning methods

1. AlexNet: is the first large-scale CNN model, which was initially created to classify millions of images in 1000 categories in ImageNet datasets [30]. The model takes input images of size 224×224 RGB (Red Green Blue) and consists of eight layers, including five convolutional layers and three fully connected (FC) layers. The AlexNet model has around 61 million parameters (Table 1). The output layer in the AlexNet model predicts the probability of images belonging to each pollen species category. This approach uses ReLU activation function, Dropout, and data enhancement strategies to avoid overfitting.
2. VGG-16: Visual Geometry Group (VGG) introduced by the University of Oxford. VGG-16 consists of 16 convolutional layers, five max-pooling layers, and three fully connected layers [13]. VGG-16 has over 138 million parameters and uses ReLU activation function and dropout regularization to improve generalization error and prevent overfitting. The final layer of VGG-16 uses the softmax activation function followed by the ReLU activation function. Images with a fixed size of 224×224 are used as inputs, and the stride is set to 1 (Table 1). The main difference compared to previous models is the deeper architecture, which includes associated double or triple convolution layers. In our model, we used the Adam optimizer with a learning rate of 0.0001, and training was performed with a batch size of 32 in 14 epochs.
3. MobileNet-V2: MobileNet-V2 is a family of neural network architectures for efficient on-device image classification and related tasks. The "V2" indicates that it's the second version of the MobileNet architecture, which includes several enhancements over the original MobileNet. The enhancements focus on improving accuracy and reducing computational complexity, making the model more efficient for mobile and edge devices where computational resources are limited. This architecture was intro-

Table 1 Properties of our scratch model and eleven pre-trained CNNs

Models	Depth (# layers)	Number of parameters (millions)	Input image size	Complexity	Speed
Scratch-model	6	5.6	224 × 224	Low	High
AlexNet	8	61	227 × 227	Low	High
VGG-16	16	40.1	224 × 224	High	Low
MobileNet-V2	53	3.5	224 × 224	Low	High
ResNet-18	18	11.4	224 × 224	Low	Low
ResNet-34	34	21.5			
ResNet-50	50	23.5			
ResNet-101	101	42.5			
ResNeSt-50	50	22.9	224 × 224	High	Low
ResNeSt-101	101	86.74			
Se-ResNeXt	101	28	240 × 240	High	High
ViT	24	26	16 × 16	High	High

duced by a team of Google engineers [31]. The MobileNet-V2 is a lightweight CNN model with 5.3 million parameters, making it remarkably efficient compared to other architectures in this study. It contains 53 layers, including an initial fully convolutional layer with 32 filters, followed by 19 residual bottleneck layers, and takes input images with a size of 224×224 (Table. 1). MobileNet-V2 architecture features linear bottlenecks between the layers and shortcut connections between the bottlenecks, enabling faster training and better accuracy. The MobileNet-V2 architecture utilizes depth-wise separable convolutions, resulting in models that are smaller, low-latency, and low-power. The use of global hyperparameters in this architecture optimizes accuracy, and the model builder can choose the most suitable model size to achieve better accuracy. Moreover, MobileNet-V2 uses 3×3 depth-wise separable convolutions that require 8 to 9 times less processing than traditional convolutions, with negligible loss in model performance.

4. ResNet (Residual Network): The ResNet architecture was developed by Microsoft researchers [32] and consists of various ResNet models, including ResNet-18, ResNet-34, ResNet-50, ResNet-101, ResNet-152, ResNet-1202, and others. In this study, we utilize ResNet-18, 34, 50, and 101. The ResNet architecture introduces a novel identity shortcut connection that skips one or more layers, which helps address the issue of vanishing gradients commonly encountered in DL models. This is especially important since using a high number of layers in transfer learning often leads to the derivatives disappearing in the network. Instead of fully connected layers, ResNet uses global average pooling. Batch normalization is also utilized in the fully connected layers to achieve convergence and enable the use of higher learning rates, leading to faster training speed. The input images for this architecture need to be of size 224×224 pixels, as shown in Table 1.
5. ResNeSt: This term is short for "ResNet with Split-attention Networks". It is an architecture developed by Facebook researchers [33] that includes between 27 and 48 million parameters. ResNeSt is a variant of ResNet that combines channel-wise attention with multi-path representation in a Split-Attention block, allowing attention across feature-map groups. Two main variants of ResNeSt are ResNeSt-50 and ResNeSt-101, which are pre-trained on the ImageNet dataset. This architecture uses an average pooling layer with a kernel size of 3×3, and input images of size 224×224 pixels (Table 1). To prevent overfitting, a dropout regularization with a probability of 0.2 is employed.
6. SE-ResNeXt: is an extension of the ResNeXt (ResNet with Next-gen architecture, it is a variant of the original ResNet model, which incorporates "next generation" enhancements for better performance) architecture that incorporates a squeeze and excitation (SE) block. It was introduced by Hu et al. [34] and contained over 28 million parameters. SE-ResNeXt uses the same basic building block as ResNeXt, which is a split-transform-merge strategy that enables parallel feature extraction. In this architecture, a squeeze and excitation (SE) block is used at the end of each non-identity branch of the residual block. The SE block performs channel-wise feature recalibration by explicitly modeling interdependencies between channels. This architecture creates a well model for several complex image datasets by stacking SE blocks together. The input image size for this model is fixed at 224×224 (Table 1).

7. Vision Transformer (ViT): is a novel image classification model that uses the Transformer architecture, which was initially developed for Natural Language Processing (NLP), over patches of images [35]. The Transformer is a deep neural network based on the attention mechanism that achieved state-of-the-art results in NLP tasks. This success has inspired computer vision researchers to use the Transformer approach for image classification tasks [36]. Unlike CNNs, which take pixels in images as input data, ViT divides the images into fixed-size patches (usually 16×16) and embeds each patch while retaining its positional embedding as input to the transformer encoder. The ViT employs self-attention to enable the model to embed knowledge across the image.

In Table 1, we compare different models in terms of their depth, parameters, input image size, complexity, and speed. Regarding the definition of complexity and speed, Complexity refers to the computational complexity of the model, which we determine primarily based on the number of layers and the number of parameters the model contains. A 'low' complexity model is one that is relatively simpler and requires fewer computational resources, typically having fewer layers and parameters. On the other hand, a 'high' complexity model is more intricate, having a higher number of layers and parameters, and thus requires more computational resources.

Regarding the speed, it refers to the inference speed of the model, which is the rate at which the model can process input and generate output. This rate is measured in terms of the number of input samples processed per unit of time. A 'high' speed model can process a larger number of input samples in a given time frame, while a 'low' speed model processes fewer.

These categorizations are relative and meant to provide a broad comparison across different models based on the various factors such as batch size, hardware accelerators (GPUs, TPUs) and software optimizations.

Experimental design and optimization techniques

The experiments for the scratch model, AlexNet, and VGG-16, data preprocessing, and analysis were conducted on a Dell Alienware (m17 R4) laptop using the Python programming language (version 3.10.6) and several unique libraries for running DL models. For other transfer learning experiments, we utilized the Microsoft Azure cloud computing platform with the Azure automated ML service, utilizing a Standard_NC6 virtual machine, GPU device (NVIDIA Tesla K-80), six cores, 56 GB RAM, and 380 GB storage. To optimize our models, we implemented several hyperparameters, including early stopping (using the Bandit policy with a slack factor of 0.1) and 15 ensemble iterations. Additionally, we utilized grid search to find the optimal hyperparameters, specifying the grid sampling method for sweeping over the defined parameter space. We set the maximum number of configurations to sweep to 100 iterations.

Performance metrics

Figure 4 shows a flowchart of the pollen classification steps using CNNs, including Input images, preprocessing steps, transfer learning models, and evaluation Metrics.

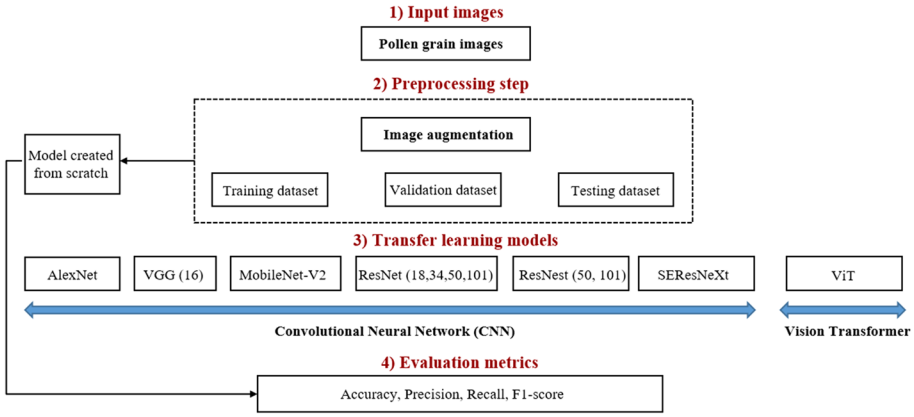

Fig. 4 Flowchart showing the pollen image classification process across several steps, including: (1) Input images; (2) Preprocessing step; (3) Transfer learning models; (4) Evaluation Metrics

This section evaluates the performance of various transfer learning models in classifying pollen grain images. The models are assessed based on accuracy, precision, recall, and F1-score. The evaluation uses a macro-average, which considers the overall study and assigns equal weight to each pollen species class. The macro-average is preferred because the dataset is relatively imbalanced, and all classes are equally significant. To analyze the experimental results, the confusion matrix is used, which provides guidance for the four outcomes: TP (True Positive), TN (True Negative), FP (False Positive), and FN (False Negative). These metrics used in this study provide insights into how well the model performs across all classes.

1. Accuracy estimates the ratio of correct predicted classes to the entire number of samples evaluated.

$$\textbf{Accuracy}: \frac{TN + TP}{TN + FN + TP + FP} \tag{1}$$

2. Recall (Sensitivity) is used to estimate the fraction of positive patterns that are accurately classified.

$$\textbf{Recall}: \frac{TP}{TP + FN} \tag{2}$$

3. Precision (Specificity) is used to estimate the positive patterns that are correctly predicted by all predicted patterns in a positive class.

$$\textbf{Precision}: \frac{TP}{TP + FP} \tag{3}$$

4. F1-score incorporates the precision and recall of a classifier into a single metric by using their harmonic mean.

$$\textbf{F1-score}: \frac{2 * Precision * Recall}{Precision + Recall} \tag{4}$$

Results and discussion

Classifying small datasets in computer vision is challenging and has been a topic of interest for many researchers [37, 38]. In our study, we addressed the issue of having a limited number of pollen images by comparing the performance of a scratch model and eleven transfer learning models. Our research builds upon a few automated classification methods for pollen grains that were developed using small datasets [19, 20, 22, 23, 25]. In this study, we trained a CNN model from scratch with six layers, fine-tuned the hyperparameters, and achieved an impressive accuracy of 91.87%. We also evaluated the performance of eleven transfer learning architectures on the classification of pollen grain images. Despite imbalances in the dataset, the models achieved excellent values for accuracy (ranging from 92.87 to 97.24%), precision (ranging from 93.50 to 97.89%), recall (ranging from 93.10 to 97.13%), and F1-score (ranging from 92.40 to 96.86%). The best-performing models were ResNeSt-101 and SE-ResNeXt, with accuracy values of 97.24% and 97.05%, respectively. On the other hand, AlexNet had the lowest accuracy of 91.87%. The study also found that deeper neural networks in the ResNet architecture (ResNet-101>ResNet-50>ResNet-34>ResNet-18) performed relatively better than shallower ones, indicating the importance of having more layers in the model to improve the learning of low and high-level features in pollen grain images. Table 2 and Figure 5 provide more information on each model's precision, recall, and F1 scores.

The ViT has a shorter training time but may not perform well on small datasets due to its high capacity and complex architecture. ViTs require a significant amount of data to generalize well and may be overfitted on limited data [39]. In addition, ViTs apply self-attention mechanisms to capture global dependencies in the image but may not capture fine-grained details as effectively as models that use convolutional layers, such as ResNeSt and SE-ResNeSt [40]. The ResNeSt-110 model has a deeper and wider architecture compared to the ViT model, which may contribute to its better performance in this study. The ResNeSt-110 model has 110 layers, while the ViT model has only 12 layers. Deeper architectures can capture more complex and abstract features, which may be necessary for accurately identifying pollen grains [41, 42]. Additionally, the ResNeSt-110 and SE-ResNeSt models have a wider architecture,

Table 2 Model performance of different transfer learning architectures in this study

	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
AlexNet	92.87	93.50	93.10	92.40
VGG-16	95.10	95.30	95.50	95.60
MobileNet-V2	94.78	96.04	95.68	95.74
ResNet-18	94.65	95.78	94.67	95.05
ResNet-34	95.32	94.78	94.57	94.47
ResNet-50	95.37	96.58	96.50	96.50
ResNet-110	96.84	96.86	96.28	96.43
ResNeSt-50	96.54	96.79	96.80	96.76
ResNeSt-101	*97.24*	*97.89*	*97.13*	*96.86*
SEResNeXt	97.05	97.66	97.31	97.01
ViT	95.95	95.71	95.46	95.54

The italic emphasis shows the most promising modeling performance

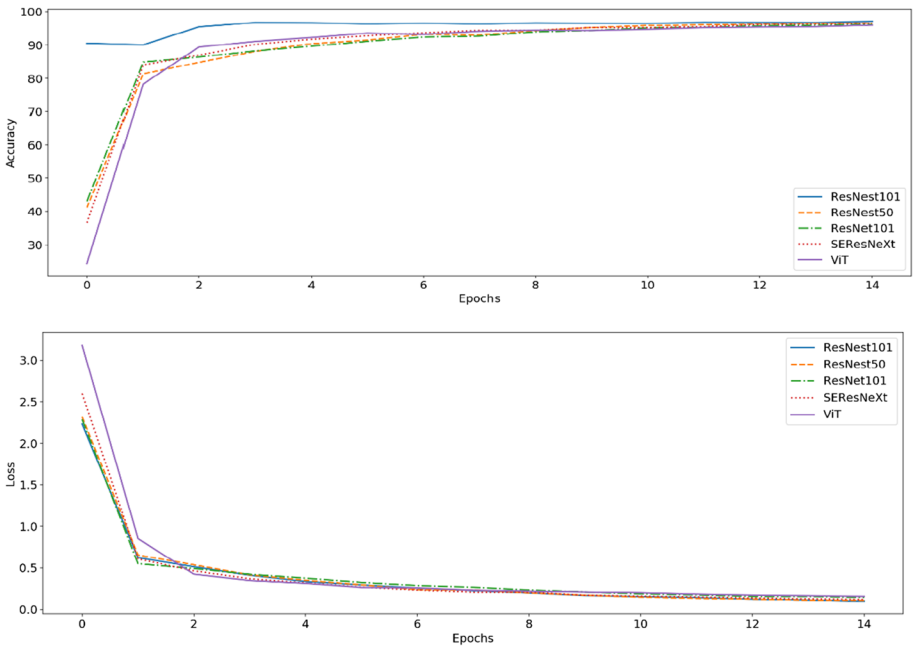

Fig. 5 The values of accuracy and loss for the top five best transfer learning models

meaning that they have more channels in each convolutional layer, which allows them to capture more diverse and informative features from the pollen grain images.

On the other hand, the MobileNet-V2 network is a lightweight model that has the smallest number of parameters, making it more suitable for use in applications with limited storage space. However, its performance on pollen classification is lower than most other architectures. Therefore, we recommend using MobileNet-V2 architecture when high classification performance is not critical, such as when the model is used in a phone application. The slight decrease in classification accuracy can be tolerable in such cases.

ResNeSt and SE-ResNeXt architectures leverage multi-scale features in a nested way, which enables them to capture more complex patterns and features in the data. Specifically, ResNeSt uses a multi-scale group convolutional approach that divides the channels of each convolutional layer into groups and aggregates them hierarchically [33]. This allows ResNeSt to capture fine-grained details in the image and learn more discriminative features, which can lead to higher accuracy and precision. At the same time, the stacked blocks in SE-ResNeXt generate a highly effective model for the pollen grain dataset [34]. The SE-ResNeXt architecture is designed to enhance the representational power of a network by allowing dynamic channel-wise feature recalibration [34].

This study also found that increasing the number of layers is useful, especially in ResNet networks. However, increasing the number of channel groups in ResNeSt and SE-ResNeXt was more effective than increasing the depth. These techniques can enhance the accuracy without increasing the parameter complexity and simultaneously reduce the number of parameters. In conclusion, our analysis highlights the limitations of using relatively shallow and simple models such as Alexnet, VGG, ResNet-18, and ResNet-34 for the pollen classification task. Our experimental results demonstrate that models with shortcut connections and Squeeze-and-Excitation networks, such as

ResNeSt and SE-ResNeXt, outperform the other models on the pollen dataset. Therefore, we recommend the use of these more advanced models for improved accuracy and performance in the context of pollen classification (Fig. 6).

Conclusion

In the context of pollen grain classification, transfer learning allows using pre-trained models on large image datasets to classify pollen grains more efficiently. With the approaches outlined here, we have demonstrated that we could achieve accurate classification results by fine-tuning a pre-trained model on a small dataset of pollen grain images while reducing the training time and computational resources required. This study demonstrated that increasing the complexity and depth of neural networks effectively achieves reliable and efficient classification of pollen grains at low taxonomic levels. However, classifying pollen grain datasets using machine learning and deep neural networks is challenging due to the relatively small and imbalanced sets of images.

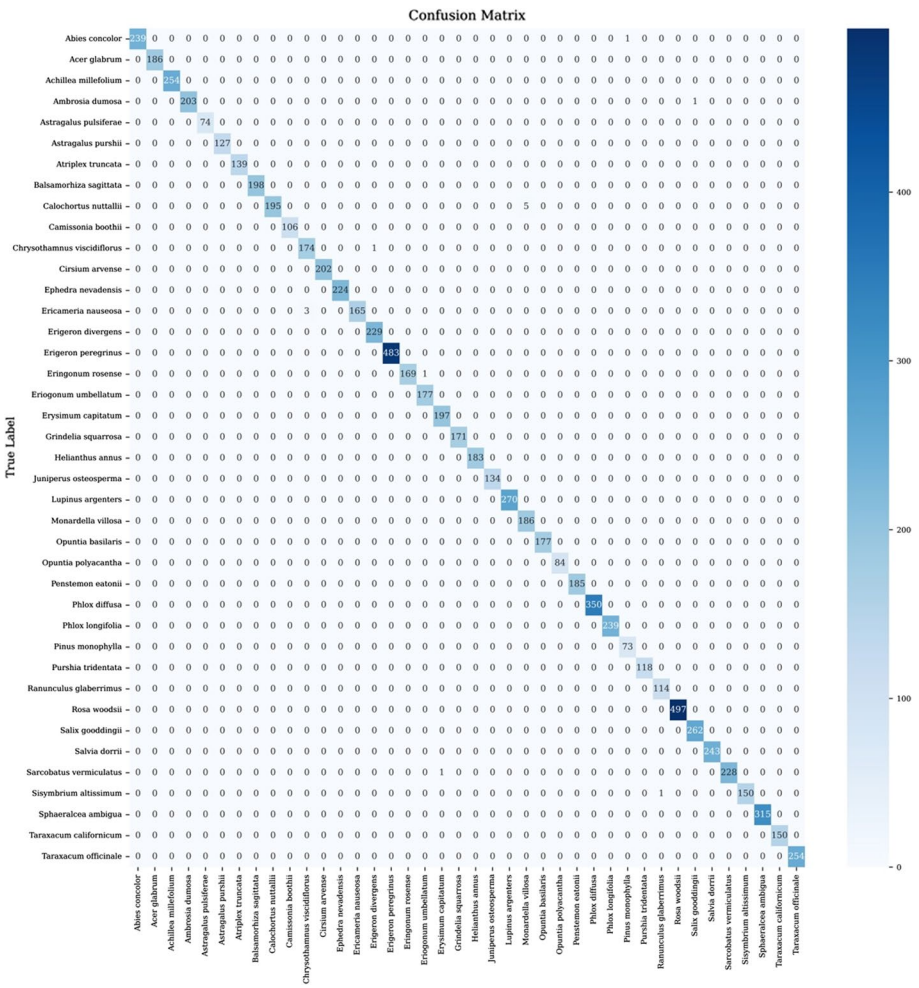

Fig. 6 Confusion matrix for the 40 pollen species used for the training dataset pollen images from the Great Basin. Rows are species identities, and columns are CNNs species assignments. The color bar indicates frequency, with dark green being the most frequent. The diagonal elements are the frequency of correctly classified outcomes, while misclassified outcomes are on the off-diagonals

Among the transfer learning techniques used, ResNeSt-101 and SE-ResNeXt performed exceptionally well, even though the CNN architecture utilized data from the ImageNet dataset, which has no image data similar to pollen grains. These techniques worked well because of their ability to capture multi-scale features, their deeper and wider architecture, and their suitability for the task of pollen grain classification.

The findings of this research have significant implications for the study of the Great Basin Desert, as identifying pollen grains can provide insights into the plant species present in the region, their distribution, and their interactions with other organisms. Further research is needed to address the limitations of the proposed models, including a focus on potential bias in the dataset and improved interpretability of the model.

Acknowledgements
We express our gratitude to the University of Nevada, Reno, Museum of Natural History for preserving and making their valuable plant collection available, which was crucial to our research. We would also like to extend our thanks to the dedicated individuals at Microsoft's AI for Earth program for granting us access to virtual machines on Microsoft Azure, which facilitated the training of our models.

Author contributions
MAR and BB were responsible for formulating the research questions, selecting appropriate DL algorithms, conducting data analysis, interpreting the results, and critically reviewing the paper. They also conducted the research and wrote the main manuscript text, including results and figures. All authors reviewed the manuscript, and all authors have read and approved the final version of the manuscript.

Funding
This work was supported by the National Science Foundation (DEB 2114942) and the University of Nevada, Reno.

Availability of data and materials
There are no restrictions on the availability of image datasets, and the authors are willing to provide them on GitHub.

Code availability
A Git repository containing all the code for this project will be shared upon acceptance.

Declarations

Ethics approval consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare that they have no known competing.

Received: 11 December 2022 Accepted: 18 August 2023
Published online: 01 October 2023

References

1. Alotaibi SS, Almeida TA. A survey of deep learning techniques for plant pollen classification. Artif Intell Rev. 2021;54(5):3937–62.
2. Zeng X, Zhang L, Chen B, Zhao Q, Zhang W, Li C, Zhang Y. Deep-learning-based palynology: applications in paleoclimatology and paleoecology. J Geophys Res Biogeosci. 2021;126(4): e2020JG005946.
3. Liu B, Huang J, Huang Y, Zhang J. Deep learning for pollen classification in forensic palynology: a systematic review. Forensic Sci Int. 2021;318: 110687.
4. Jaccard P, Cosandey-Godin A, Pernet L, Rey P, Guisan A. Improving the automation of pollen identification: a deep learning approach. Appl Plant Sci. 2020;8(9): e11372.
5. Borkhataria R, Bhandari S, Bhat A, Mala S. Automated pollen identification: an evaluation of the performance of machine learning algorithms. For Sci Int. 2016;266:426–33.
6. Chevallier E, De Beaulieu JL. Quantitative pollen-based climate reconstruction: a critical analysis of approaches, methods, and techniques. Quatern Sci Rev. 2011;30(27–28):3934–48.
7. Sevillano V, Holt K, Aznarte JL. Precise automatic classification of 46 different pollen types with convolutional neural networks. PLoS ONE. 2020;15(6): e0229751.
8. Wäldchen J, Mäder P. Plant species identification using computer vision techniques: a systematic literature review. Arch Comput Methods Eng. 2018;25(2):507–43.
9. Buddha K, Nelson H, Zermas D, Papanikolopoulos N. Weed detection 401 and classification in high altitude aerial images for robot-based precision 402 agriculture. In: 27th Mediterranean Conference on Control and Automation 403 (MED); 2019. p. 280–285.

10. Afonso M, Fonteijn H, Fiorentin FS, Lensink D, Mooij M, Faber N, Polder G, Wehrens R. Tomato fruit detection and counting in greenhouses using deep learning. Front Plant Sci. 2020;11:571299.
11. Norouzzadeh MS, Morris D, Beery S, Joshi N, Jojic N, Clune J. A deep active learning system for species identification and counting in camera trap images. Methods Ecol Evol. 2021;12(1):150–61.
12. Balmaki B, Rostami MA, Christensen T, Leger EA, Allen JM, Feldman CR, Forister ML, Dyer LA. Modern approaches for leveraging biodiversity collections to understand change in plant-insect interactions. Front Ecol Evol. 2022;10: 924941.
13. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv preprint; 2014. http://arxiv.org/abs/1409.1556.
14. Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. Adv Neural Inf Process Syst. 2012;25:1097–105.
15. Goodfellow I, Bengio Y, Courville A. Deep learning, vol. 1. MIT Press; 2016.
16. Shorten C, Khoshgoftaar TM. A survey on image data augmentation for deep learning. J Big Data. 2019;6:60.
17. Yosinski J, Clune J, Bengio Y, Lipson H. How transferable are features in deep neural networks? Adv Neural Inf Process Syst. 2014;27:3320–8.
18. Zhang K, Liu Z, Shen Y. A survey on transfer learning for image classification. J Vis Commun Image Represent. 2020;69: 102795.
19. Polling M, Li C, Cao L, et al. Neural networks for increased accuracy of allergenic pollen monitoring. Sci Rep. 2021;11(1):11357–67.
20. Chauhan S, Vig L, De Filippo De Grazia M, Corbetta M, Ahmad S, Zorzi M. A comparison of shallow and deep learning methods for predicting cognitive performance of stroke patients from MRI lesion images. Front Neuroinform. 2019;13:53.
21. Tan J, Li Y, Chen H, Zhou F. Deep learning for image-based pollen recognition: a review. Micromachines. 2018;9(9):454.
22. Daood A, Ribeiro E, Bush M. Pollen grain recognition using deep learning. In: Advances in visual computing. Lecture notes in computer science. Springer: Cham; 2016. p. 321–30.
23. Vedaldi A, Lenc K. MatConvNet-convolutional neural networks for MATLAB. Computer vision and pattern recognition; 2014. http://arxiv.org/abs/1412.4564v3.
24. Khanzhina N, Putin E, Filchenkov A, Zamyatina E. Pollen grain recognition using convolutional neural networks. In: ESANN; 2018.
25. Sevillano V, Aznarte JL. Improving classification of pollen grain images of the polen23e dataset through three different applications of deep learning convolutional neural networks. PLoS ONE. 2018;13(9):1–18.
26. de Geus AR, Batista MA. Large-scale pollen recognition with deep learning. In: 27th European signal processing conference (EUSIPCO); 2019.
27. Olsson O, Karlsson M, Persson AS, Smith HG, Varadarajan V, Yourstone J, Stjernman M. Efficient, automated and robust pollen analysis using deep learning. Methods Ecol Evol. 2021;12:850–62.
28. Astolfi G, Gonçalves AB, Menezes GV, Borges FSB, Astolfi ACMN, Matsubara ET, Alvarez M, Pistori H. POLLEN73S: an image dataset for pollen grains classification. Ecol Inf. 2020;60: 101165.
29. Kubera E, Kubik-Komar A, Piotrowska-Weryszko K, Skrzypiec M. Deep learning methods for improving pollen monitoring. Sensors. 2021;21:3526.
30. Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. In: Pereira F, Burges CJC, Bottou L, Weinberger KQ, editors. Advances in neural information processing systems 25. Curran Associates, Inc; 2012. p. 1097–105.
31. Sandler M, Howard A, Zhu M, Zhmoginov A, Chen L. Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2018. p. 4510–20.
32. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. arXiv preprint. 2015; http://arxiv.org/abs/1512.03385. p. 32.
33. Zhang H, Chongruo W, Zhang Z, Zhu Y, Zhang Z, Lin H, Sun Y, He T, Mueller J, Manmatha R, Li M, Smola A, ResNeSt: split-attention networks. In: Computer vision and pattern recognition. IEEE; 2020.
34. Hu J, Shen L, Sun G. Squeeze-and-excitation networks. In: Conference on computer vision and pattern recognition. IEEE; 2018. 7132–41.
35. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN. et al. Attention is all you need. Advances in neural information processing systems (Long Beach, CA); 2017. p. 5998–6008.
36. Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, Dehghani M, Minderer M, Heigold G, Gelly S et al. An image is worth 16x16 words: transformers for image recognition at scale. arXiv; 2020; http://arxiv.org/abs/2010.11929v2.
37. Delgado JMD, Oyedele L. Deep learning with small datasets: using autoencoders to address limited datasets in construction management. Appl Soft Comput. 2021;112: 107836.
38. Hasan K, Alam A, Dahal L, Roy S, Wahid SR, Elahi TE, Marti R, Khanal B. Challenges of deep learning methods for COVID-19 detection using public datasets. Inform Med Unlocked. 2022;3: 100945.
39. Touvron H, Vedaldi A, Douze M, Jégou H. Training data-efficient image transformers & distillation through attention. In: Advances in Neural Information Processing Systems; 2021. (NeurIPS).
40. Hassani K, Huang T. Escaping the big data paradigm with compact transformers. In: International conference on learning representations (ICLR); 2021.
41. Huang G, Liu Z, Van Der Maaten L, Weinberger KQ. Densely connected convolutional networks. In: IEEE conference on computer vision and pattern recognition (CVPR); 2021.
42. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, et al. Going deeper with convolutions. In: IEEE conference on computer vision and pattern recognition (CVPR); 2015.

Publisher's Note