JGR Solid Earth

Author Contributions:

Conceptualization: Chetan L. Nathwani
Methodology: Chetan L. Nathwani, William Brownscombe, Cédric M. John
Software: Chetan L. Nathwani
Supervision: Jamie J. Wilkinson
Visualization: Chetan L. Nathwani
Writing – original draft: Chetan L. Nathwani
Writing – review & editing: Jamie J. Wilkinson, William Brownscombe, Cédric M. John

Mineral Texture Classification Using Deep Convolutional Neural Networks: An Application to Zircons From Porphyry Copper Deposits

Chetan L. Nathwani[1,2,3] [iD], **Jamie J. Wilkinson[1,2], William Brownscombe[1], and Cédric M. John[2]** [iD]

[1]Department of Earth Sciences, Natural History Museum, London, UK, [2]Department of Earth Science and Engineering, Imperial College London, South Kensington Campus, London, UK, [3]Now at Department of Earth Sciences, Institute of Geochemistry and Petrology, ETH Zurich, Zurich, Switzerland

Abstract The texture and morphology of igneous zircon indicates magmatic conditions during zircon crystallization and can be used to constrain provenance. Zircons from porphyry copper deposits are typically prismatic, euhedral, and strongly oscillatory zoned which may differentiate them from zircons associated with unmineralized igneous systems. Here, cathodoluminescence images of zircons from the Quellaveco porphyry copper district, Southern Peru, were collected to compare zircon textures between the premineralization Yarabamba Batholith and the Quellaveco porphyry copper deposit. Quellaveco porphyry zircons are prismatic, euhedral, and strongly oscillatory zoned, whereas the batholith zircons are subhedral-anhedral with weaker zoning. We adopt a deep convolutional neural network (CNN) approach to demonstrate that a CNN can classify Quellaveco porphyry zircons with high success. We trial several CNN architectures to classify zircon images: LeNet-5, AlexNet and VGG, including a transfer learning approach where we used the weights of a VGG model pretrained on the ImageNet data set. The VGG model with transfer learning is the most effective approach, with accuracy and receiver operating characteristic-area under curve (ROC-AUC) scores of 0.86 and 0.93, indicating that a Quellaveco porphyry zircon CL image can be ranked higher than a batholith zircon with 93% probability. Visualizing model layer outputs demonstrates that the CNN models can recognize crystal edges, zoning, and mineral inclusions. We trial implementing trained CNN models as unsupervised feature extractors, which can empirically quantify crystal textures and morphology. Therefore, deep learning provides a tool for the extraction of information from large, imaged-based petrographic data sets which can facilitate petrologic and provenance studies.

Plain Language Summary Zircon is a common mineral found in rocks associated with magmatism. Zircons can have a range of shapes, sizes, and internal structures which reflect the conditions in the magma chamber where the zircon crystallized. It has been suggested that zircons derived from magmas associated with porphyry copper deposits, economically valuable sources of copper, may possess specific morphologies and textures. Here, we performed textural imaging of zircons from the Quellaveco porphyry deposit in Southern Peru and neighboring granitic rocks not associated with porphyry Cu mineralization. We find the porphyry zircons have the expected characteristic shapes and textures previously proposed, whereas the other rocks do not. We trained a deep learning model to effectively classify images of zircons from porphyry Cu rocks, which is capable of recognizing the edges, zones, and inclusions within zircon crystals. Our approach can be used to identify zircons that are associated with porphyry Cu deposits, and similar methods can be used to constrain the provenance (host rock) of zircons in stream sediments.

1. Introduction

Over the past decade, artificial neural networks (ANNs) have transformed the field of machine learning and artificial intelligence, becoming the preferred approach for computer vision, speech recognition, and machine translation. They have consequently been applied across a number of disciplines, including geoscience, with the capability to automate tasks and improve performance. Neural networks have become particularly powerful at performing tasks on unstructured data such as images, video, audio, and text (Goodfellow et al., 2016; LeCun et al., 2015). Recently, advances in convolutional neural networks (CNNs) have led to the development of models that can classify images with superhuman performance (Russakovsky et al., 2015). These techniques have the potential to automate and thence improve the efficiency of a number of tasks in geoscience (Dramsch, 2020),
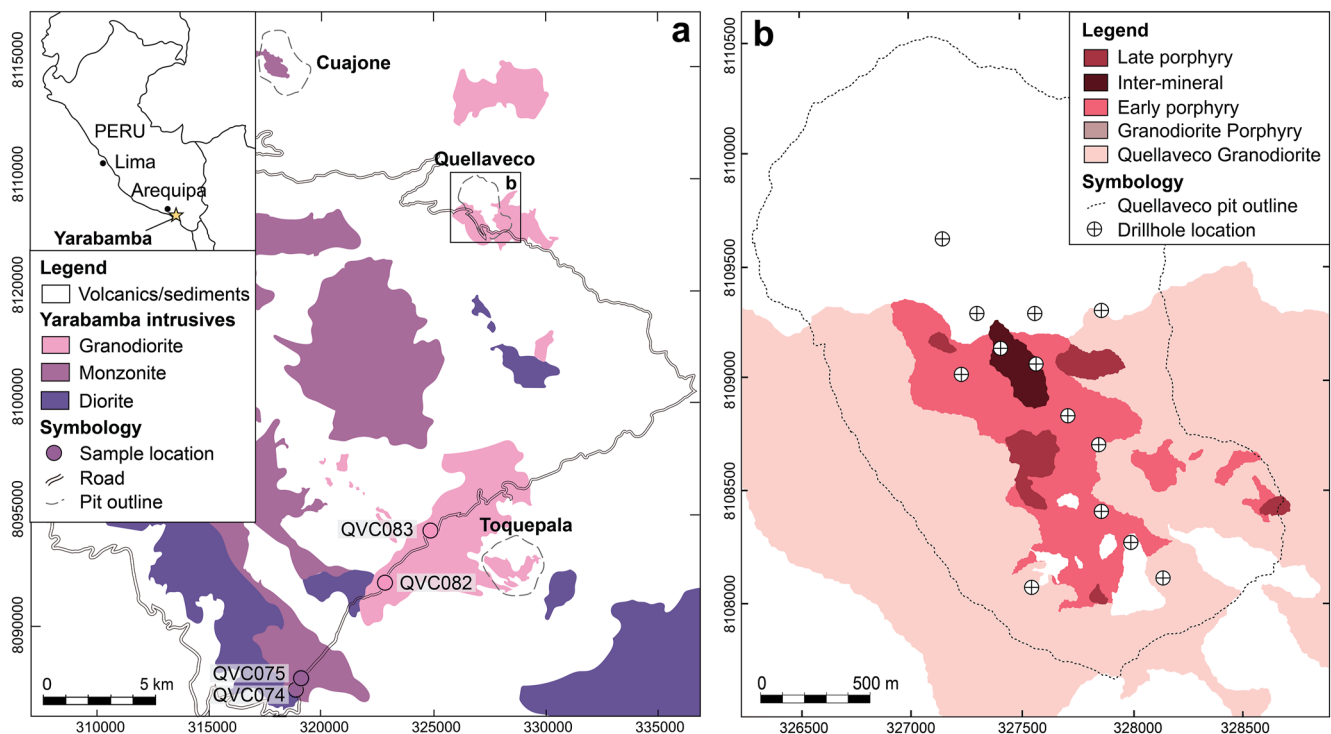
Figure 1. Simplified geological map of the (a) Quellaveco district and (b) Quellaveco deposit after Nathwani et al. (2021). Inset map in (a) shows the location of the district (star) in Peru. District-scale sample localities are shown in (a) and drillholes from which porphyry samples were taken are shown in (b). For further information on these samples please refer to Nathwani et al. (2021).

including mineral exploration routines. Several recent studies have demonstrated their value in automation of tasks in fields such as seismic interpretation (Waldeland et al., 2018), rock classification (Pires de Lima et al., 2019), ash classification (Shoji et al., 2018), landform detection (Palafox et al., 2017), petrography (Koeshidayatullah et al., 2020), and mineral exploration (Li et al., 2021; Maepa et al., 2021).

A recent focus of mineral exploration has been the development of mineral chemistry approaches to distinguish between porphyry Cu fertile igneous rocks and unfertile granitoids (Cooke et al., 2020). For example, previous work has demonstrated that zircons from porphyry Cu-related intrusions have a distinct trace element composition (Ballard et al., 2002; Dilles et al., 2015; Loader et al., 2017; Lu et al., 2016). It has also been suggested that zircons from porphyry Cu deposits may have a characteristic texture under cathodoluminescence (CL; Lu et al., 2016), such as strongly oscillatory zoned, euhedral, prismatic-pyramidal crystals with abundant mineral inclusions (Leslie et al., 2021; Pizarro et al., 2020). The presence of these properties in detrital zircons during porphyry Cu exploration could be used as an initial method to identify catchment areas or magmatic systems which are predisposed to host porphyry Cu deposits.

This paper aims to develop a deep learning methodology to recognize and classify zircon textures, and assess whether such an approach is capable of discriminating between zircon CL images from porphyry Cu mineralized rocks and premineralization granitoids in Southern Peru. Useful theory for the development of CNN architectures that can discriminate different textural populations is outlined, including model regularization, transfer learning, and feature extraction, which can be implemented in a variety of tasks across mineralogy and petrology.

2. Methods

2.1. Sample Selection and CL Imaging

Intrusive rocks from the Quellaveco porphyry Cu-Mo system, Southern Peru, were collected from the Yarabamba Batholith and from the Quellaveco porphyry intrusions. This batholith corresponds to the host rock and is an intrusion which predates the Quellaveco porphyry deposit (Figure 1). Because the Yarabamba Batholith is

composed of several spatiotemporally distinct intrusive phases (Figure 1a), three predominant intrusive phases of the Yarabamba Batholith were sampled (Nathwani et al., 2021). The oldest two phases, the Yarabamba Monzonite and Toquepala Granodiorite, were sampled from field outcrops, whereas the latest phase (the Quellaveco Granodiorite) was sampled from drill-core at the Quellaveco deposit (Figure 1b). Six samples of the batholith were selected (including two Quellaveco Granodiorite samples from drill-core) and nine drill-core samples from the Quellaveco porphyry Cu deposit. A full account of the geological context of these lithologies and the collected samples can be found in Nathwani et al. (2021).

Zircons were separated at ZirChron LLC in Arizona, USA. Samples were crushed by jaw crushing and disk pulverization, followed by heavy liquid and magnetic separation. Zircons were picked from the separates, mounted in epoxy and polished. As many zircons as possible were mounted to ensure that as wide a range of sizes, shapes, and textures were represented. To ensure no sampling bias occurred during mounting, zircon crystals were carefully poured onto grain mounts, rather than being hand-picked.

Zircons were imaged using back-scattered electron imaging and CL using a TESCAN TIMA scanning electron microscope in the Imaging and Analysis Centre (IAC) at the Natural History Museum, London. The TIMA system comprises four EDAX Element silicon drift detectors and a TESCAN MIRA 3 Schottky field emission gun SEM. Data were collected using 2 μm pixel spacing. As well as CL image acquisition, energy dispersive spectra (EDS) were also acquired per pixel, in order to check for any non-zircon grains and to monitor the occurrence of zircon-hosted mineral inclusions. The instrument was operated at 25 kV and 14 nA. The BSE, CL, and EDS data were acquired simultaneously using the high resolution mapping mode (Hrstka et al., 2018). Generally, the blocks contained almost entirely zircon crystals; however, if this was not the case, a filter could readily be applied, based on EDS data, to produce maps that only contained zircon.

The CL maps of each grain mount were split to produce one image per sample. The montaged CL images for zircons from each sample are provided in Data Set S1. These images were then segmented using a *Canny* edge detection operator (Canny, 1986) in the OpenCV library for Python (Bradski, 2000) to produce individual images for each zircon grain. This successfully isolated over 85% of the zircon crystals (750 images in total). However, infrequently, individual grains could not be separated due to overlaps between them. Such images were manually discarded. The models presented herein also required that every input image was equal in *xy* dimensions, which was not initially the case because of the large size distribution of zircon crystals. Some crystals were surrounded by empty resin, whereas others filled the entire image square. Each image was therefore automatically cropped (outer pixels removed) or padded (pixels added to outside of image) to produce uniformly sized images of 90×90 pixels. Every image was then resized to a 32×32 image appropriate for the CNN model outlined below (with 1 or 3 channels depending on the model used). The intensities of pixels were normalized to between 0 and 1 to decrease training time.

Two additional sets of zircon images (one from the premineralization batholith, and one from the porphyry) were kept aside as an independent test set for the trained models. The two samples are QVC075 (Yarabamba Monzonite) and QVC055 (Late Porphyry), which were never seen by the model during the training process. These images were preprocessed identically to the training examples.

2.2. Convolutional Neural Networks

ANNs are used to map a fixed-size input (e.g., an image) to a fixed-sized output (e.g., a class probability) using a series of interconnected neurones (Hornik et al., 1989). Each neurone consists of a weighted sum of *i* input features, which are passed through a nonlinear activation function, *f*, to produce an output, *y*. Each input feature, x_i, is multiplied by a weight, w_i, and a bias (offset), *b*, term is added

$$y(x) = f\left(b + \sum_{i=1}^{n} w_i x_i \right) \qquad (1)$$

Applying a nonlinear activation function introduces nonlinearity to the output of a neurone, allowing multiple neurones to be stacked together and thus the network can model complex nonlinear functions. The most widely used activation function is the rectified linear unit function (ReLU) which is the half-wave rectifier ($f(x) = max(z,0)$). The ReLU activation function typically leads to faster convergence of the neural network
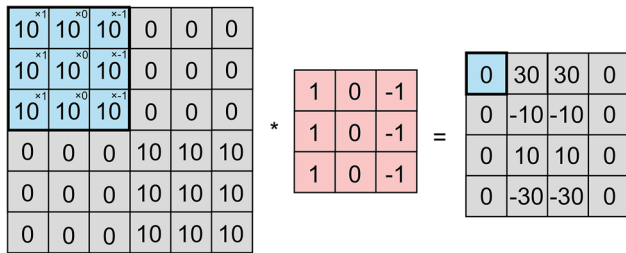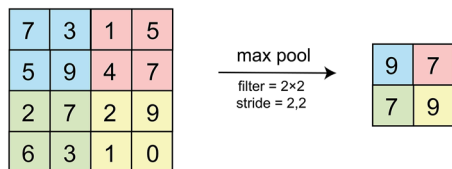
Figure 2. Schematic illustrations of two key operations in a convolution neural network: (a) the convolutional operation and (b) the pooling operation. The convolutional operation demonstrates a vertical edge detection kernel applied to an image. The derived feature map highlights the presence of a vertical edge in the center of the image. The pooling operation shows a maximum pooling operation applied to an image, with a filter size of 2×2 and a stride of 2,2.

compared to other activation functions due to the nonsaturation of its gradients (i.e., the gradient never becomes exceedingly small or large (Glorot et al., 2011; Nair & Hinton, 2010)).

Neurones are aggregated into layers, and several layers can be arranged in sequence (known as "hidden layers") through which data passes from the input to be mapped to the output. Before training of an ANN, the weight terms in the network are initialized and as each training example is passed through the network, the error rate is evaluated using a cost function. The weight terms in the network are sequentially adjusted through the network to minimize the cost function, by backpropagation and gradient descent. Adjustment of the weight in the descent direction occurs in iterations, where the step size, known as the learning rate, is determined by the user. A large learning rate will cause large changes in the weights in the direction of the descent direction (and may lead to the weight overshooting the optimum), while a small learning rate causes small changes (and may lead to slow convergence).

The application of traditional feed-forward ANNs to image and pattern recognition is limited for several reasons (LeCun & Bengio, 1998). First, an example 256×256 pixel image, which contains 65,536 pixels, would require training 65,536 weights in a single neurone in the first layer of the network. This would be extremely computationally expensive and prone to overfitting. Second, traditional ANNs are limited because they are not invariant to translation or distortion of the input. For example, classifying whether a cat is present in an image requires the model to work regardless of the position or size of the cat in the image. Furthermore, image classification must take advantage of the spatial dependency of features in an image, such as the presence of multiple whiskers above a mouth.

A subset of ANNs called CNNs are instead the most popular method for image classification. CNNs are algorithms which can assign variable importance to different features of images and use these for classification. The classic architecture of a CNN consists of three main types of layers: convolutional, pooling, and fully connected.

2.2.1. Convolutional Layers

A convolutional layer applies a filter to an image, which transforms the image to a feature map which is passed to the next layer in the network (Figure 2a). The filter is a pixel grid, e.g., a 3×3 matrix, which is applied to a 3×3 area of the image, and the dot product is calculated as an output value. The filter shifts across the image, known as the stride, eventually producing a feature map. The 3×3 filter would be composed of 9 weight terms which are adjusted during the backpropagation and gradient descent process to produce the optimal feature maps for the task. Applying the same filter across an entire image has important advantages in that it reduces the number of weights to train and it allows features to be detected in an image regardless of position (i.e., translation invariance).

2.2.2. Pooling Layers

Pooling has the key function of reducing the dimensionality of the input, thus reducing the computational power required to train the model. A kernel, such as a 2×2 grid, will move across a convolved image and either the maximum (known as max pooling) or average pixel value (average pooling) is taken from each 2×2 area (Figure 2b). A typical CNN consists of several convolutional and pooling layer pairs placed in sequence, where the earlier layers learn broader scale features (e.g., edges) and deeper layers learn more detailed features.

2.2.3. Fully Connected Layers

A fully connected layer is a layer which is typically used in a feed-forward ANN, where every neurone in one layer is connected to every neurone in the next layer. The feature grid produced from the convolutional layers and pooling layers is flattened into a vector and passed through the layer of neurones (a fully connected layer). The output of one, or more, fully connected layers is then used for classification, where a sigmoid (for binary classification) or softmax (for multiclass classification) function is applied to produce a probability.

2.3. Model Regularization

Overly complex machine learning models can learn the training data too well and therefore have poor performance (insufficient generalization) when exposed to new data. This is termed overfitting. One of the most common ways to mitigate against overfitting is weight regularization, in which a regularization term is added to the cost function—which penalizes the model if weights become too large—forcing the weights to become smaller, thus producing a less complex model (L1 and L2 regularization; Ng, 2004). More recently, dropout regularization has become a favored approach (Hinton et al., 2012; Srivastava et al., 2014), and generally performs better than weight regularization (Wager et al., 2013). Dropout is a method where randomly selected neurones are dropped out during the training process. This has the effect of reducing dependencies on neurones, causing weights to become more evenly distributed, and thus producing a less complex model. The probability of dropping out each neurone during training is defined by the probability term p. During validation and testing, dropout is not used, and the output weights are reduced (by $1 - p$) in order to compensate for the larger outputs from the layers that were dropped out during training (Srivastava et al., 2014).

Another highly regarded regularization approach for CNNs is data augmentation, where images in the training data set are slightly modified randomly to increase the diversity of the training data set (Perez & Wang, 2017; Shorten & Khoshgoftaar, 2019). Examples of transformations that can be applied are: rotation, mirroring, zooming, changing contrast, and brightness. Data augmentation can be performed using preprocessing layers in the Python implementation of *keras* (Chollet, 2015), which enables this step to be integrated within the CNN pipeline so that for every training epoch, each image in the training set undergoes a random data augmentation.

A final useful regularization approach is batch normalization (Ioffe & Szegedy, 2015). During each iteration of adjusting parameters during a training epoch (known as a mini-batch), the activations for each layer are normalized to a fixed mean and variance. The main benefit of this is that it causes an acceleration in the training process, since the variance of the weights is reduced which leads to a faster convergence during gradient descent (Ioffe & Szegedy, 2015). Batch normalization can also have a regularizing effect since it adds noise to the activations and thus can reduce overfitting.

2.4. Model Selection

Selecting an appropriate CNN for a classification task can be challenging because there are a huge number of parameters that can be varied, such as the number of layers, the kernel size, and stride (the number of pixels moved by the kernel after each step) of convolutional and pooling layers and the number of neurones in fully connected layers. This study follows the approach of adapting preexisting model architectures which have proved successful across numerous domains. The earliest CNN model, known as LeNet-5, which was proposed for the recognition of handwritten digits (LeCun et al., 1989, 1998), consists of two pairs of convolutional and pooling layers followed by three fully connected layers. More recent CNN models have employed more sophisticated architectures and a larger number of layers: e.g., the GoogLeNet model (Inception v1; Szegedy et al., 2014) contains 22 layers and millions of trainable parameters. In this study, four different CNN architectures were used, to compare the relative performance of models with increasing numbers of parameters and complexity:

1. A baseline neural network model with no convolution or pooling operations (i.e., a multilayer perceptron).
2. An adaptation of the LeNet-5 CNN model originally proposed by LeCun et al. (1989, 1998), with some modifications based on more recent CNN model advances.
3. The AlexNet CNN model (Krizhevsky et al., 2012), originally proposed for classification of 1.2 million high resolution images belonging to 1,000 classes (the ImageNet database; Deng et al., 2009).
4. The VGG model (Simonyan & Zisserman, 2015), also originally designed for classification of the ImageNet database, which contains 16–19 layers.

A summary of these models can be found in Table 2, in the Supporting Information S1 (Tables S1–S4) and in the following subsections. The ANN which does not contain any convolutional or pooling layers was trained for comparative purposes. This baseline model flattens the input image and contains two hidden layers with 128 and 32 neurones, respectively.

2.4.1. A Modified LeNet Architecture

Here, an adaptation of the LeNet-5 model is first applied, since the model was designed for classification of gray-scale images with smaller size, whereas recent CNN models that were designed for classification of color images
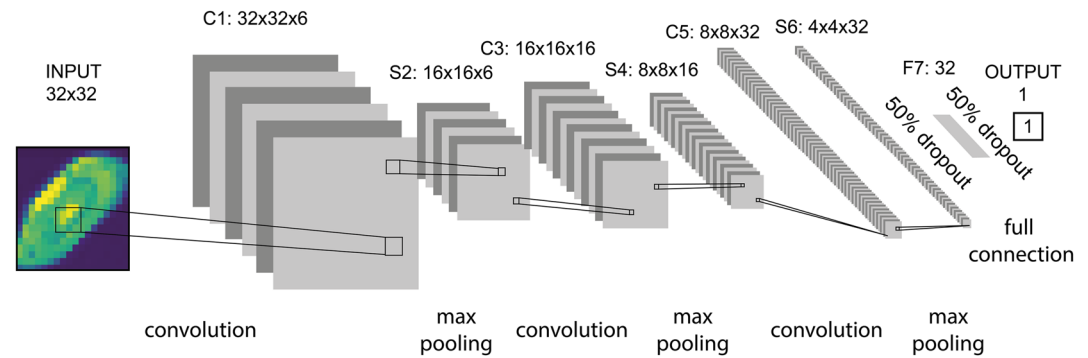
Figure 3. Schematic illustration of the LeNet-style convolutional neural network used in this study. Repeated sequences of convolution (C), max pooling (S) operations are applied to decrease the size of the image, while extracting the key features from the image. After each step, the dimension of the image is shown (width, height, channels). After the final max pooling layer, the image is flattened to produce a 512-dimensional vector, from which two fully connected layers (F) are used to predict the binary output. Between these F layers, dropout regularization is used with a probability of 0.5.

of larger size. Five modern advances in CNN architecture have been added to the LeNet-5 model to improve performance:

1. 3×3 convolutions were used instead of 5×5 convolutions, since these have a smaller receptive field and therefore can capture more complex features such as oscillatory zoning. This also reduces the number of trainable parameters by ~25,000.
2. Padding was used, where the images are padded with layers of zeros prior to convolution operations. This mitigates against the undesirable effect of convolutions causing pixels at the edge and corners of images to be used much less than pixels at the center, which may lead to information loss. Padding was only used in the first convolutional layer because, as the image dimensions reduces, information becomes more uniformly distributed across the image. Padding retains the height and width of images after a convolution has been applied.
3. Use of ReLU activation functions, as opposed to the *tanh* functions used in the original LeNet-5 model. The ReLU activation function has become the preferred option in neural networks (Glorot et al., 2011; Krizhevsky et al., 2012; Nair & Hinton, 2010) because the logistic and *tanh* functions suffer from vanishing gradients (i.e., gradient descent struggles to converge because gradients can become very small).
4. Max pooling was used instead of average pooling because the former has been shown to have stronger performance in object detection (Ranzato et al., 2007; Scherer et al., 2010).
5. Regularization approaches have been implemented to reduce the probability of model overfitting. Dropout regularization was applied to the CNN models developed in this study, with a dropout probability of 0.5 set during the training process. Data augmentation was performed in order to ensure that the CL images in the training data set represented the continuum of possibilities that may be encountered. Random rotation (0.2), zooming (0.4), and modification of brightness and contrast (0.2) of images were used (Figure S1 in Supporting Information S1). The latter two methods are particularly important since, during image acquisition, brightness, and contrast were empirically set and therefore models could be biased toward classifying by these criteria.

The LeNet architecture used here consists of seven hidden layers (Figure 3). The first convolutional layer C1 applies 6 convolutions of 3×3, using a stride of 1, to produce 6 feature maps of 32×32. A pooling layer S2, uses max pooling to halve the number of pixels in the feature maps, hence outputting 6 feature maps of size 16×16. A second convolutional layer, C3, applies 16 3×3 convolutions using a stride of 1 to produce 16 feature maps of size 16×16. This is then max pooled by pooling layer S4 to produce an 8×8 image. The final convolutional layer C5 applies 32 3×3 convolutions to produce 32 8×8 images. The final pooling layer S6 then produces 32 4×4 images. These images are flattened to form a one-dimensional pixel vector of 512 length. A fully connected layer, F7, is then used to map this into a 32-feature vector. A dropout layer is used in between with a dropout probability of 0.5. A final output layer, along with another dropout layer ($p = 0.5$) is used and, in the final layer, a sigmoid activation function is used to produce a probability between 0 and 1 for the binary output (i.e., whether the zircon image is derived from a porphyry Cu-related rock).

Overall, four versions of the LeNet-style architecture model were run, with differing components of regularization, in order to evaluate the contribution of these methods to improving performance. The four LeNet-style

models trained are: (a) no regularization; (b) dropout; (c) data augmentation; and (d) both. Batch normalization (Ioffe & Szegedy, 2015) was also tested, although this was found not to improve model performance.

2.4.2. AlexNet

Many of the more recently developed CNN architectures are likely too large and complex for this task given they are designed for large RGB images (e.g., $224 \times 224 \times 3$) and for multiclass classification purposes (e.g., 1,000 classes). After the introduction of LeNet-5, Krizhevsky et al. (2012) introduced a CNN architecture, known as AlexNet. This model contains five convolutional layers, three max pooling layers, and three fully connected layers. The model uses ReLU activation functions, data augmentation, and dropout.

A modified version of AlexNet was implemented, with two additional, fully connected layers to reduce the output to a binary classification. This model contains 10.3 million trainable parameters. Batch normalization was used after each convolutional layer to speed up the training process.

2.4.3. VGG and Transfer Learning

The VGG-16 CNN architecture contains 16 layers (13 convolutional layers and 3 pooling layers), and the convolutional layers use small 3×3 convolutions (Simonyan & Zisserman, 2015). The model was originally built for classification of the ImageNet database (i.e., high resolution, multiclass images) and so a modified classification head was added to produce a binary output. The images here ($64 \times 64 \times 1$) were padded with zeros and converted to three channels (by duplicating the singular color channel to three channels) to meet the input dimension size for the model ($224 \times 224 \times 3$). In total, this model contains 14.8 million trainable parameters.

Due to the large size of this model, transfer learning was used in order to reduce training time and to test whether it may improve model performance. Transfer learning is when the weights of a pretrained model on a larger data set are used, and only the classification layer and possibly a few top layers of the CNN are trained from scratch. A common example, which is used here, is when large CNN models are being trained for image classification tasks; the weights can be initialized using pretrained weights on the ImageNet data set. The classification head that was added to this model consists of three fully connected layers which maps a 512-dimensional output to 128, 32 and finally a binary output.

2.4.4. Model Training

All models were trained using the Adam (Adaptive Moment Estimation) optimization algorithm for optimizing the minimum of the cost function (Kingma & Ba, 2017). A mini-batch size of 32 was used, meaning that weights were updated after 32 training examples had been passed through the network. These iterations continue until the entire training set has undergone a forward pass through the network, which is known as a training epoch. A learning rate of 0.001 was used (except for the VGG model), which determines the step size for each iteration of the training process.

During training of the VGG model, the VGG base model was frozen, so only the classification head was trained (69,825 trainable parameters), for 200 epochs with a learning rate of 0.001 (Figure 7). The model was then "fine-tuned" by unfreezing the final convolutional layer (a total of 2,429,633 trainable parameters) of the VGG model, and training for 20 more epochs at a lower learning rate (0.0001). Training the model for more epochs and/ or using a higher learning rate led to overfitting. Unfreezing additional layers of the VGG model was not found to further improve performance.

In all models, the cost function used was the binary cross-entropy loss function. During an epoch, each training example underwent data augmentation, where the image was randomly flipped, rotated, and contrast and brightness adjusted. The accuracy, loss, and the receiver operating characteristic-area under curve (ROC-AUC) were evaluated after each training epoch. The focus was to train the models to optimize the ROC-AUC, since the ability to rank probabilities of porphyry zircons higher than batholith zircons is the most desirable metric if such models are to be applied to mineral exploration tasks. This metric is also better suited to class imbalanced data sets.

The CNN models in this study were built and applied using open-source machine learning packages for Python (Tensorflow v.2.6.0; Abadi et al., 2016 and Tensorflow-Keras v. 2.4.0; Chollet, 2015). Models (with exception of the LeNet model due to its small number of trainable parameters) were trained using a GPU to accelerate the training process. The training was run on Google Cloud (Google Colabatory) using a NVIDIA T4 GPU. The Python3 code used for this study is available in annotated form as a Jupyter Notebook (see Data Availability Statement).
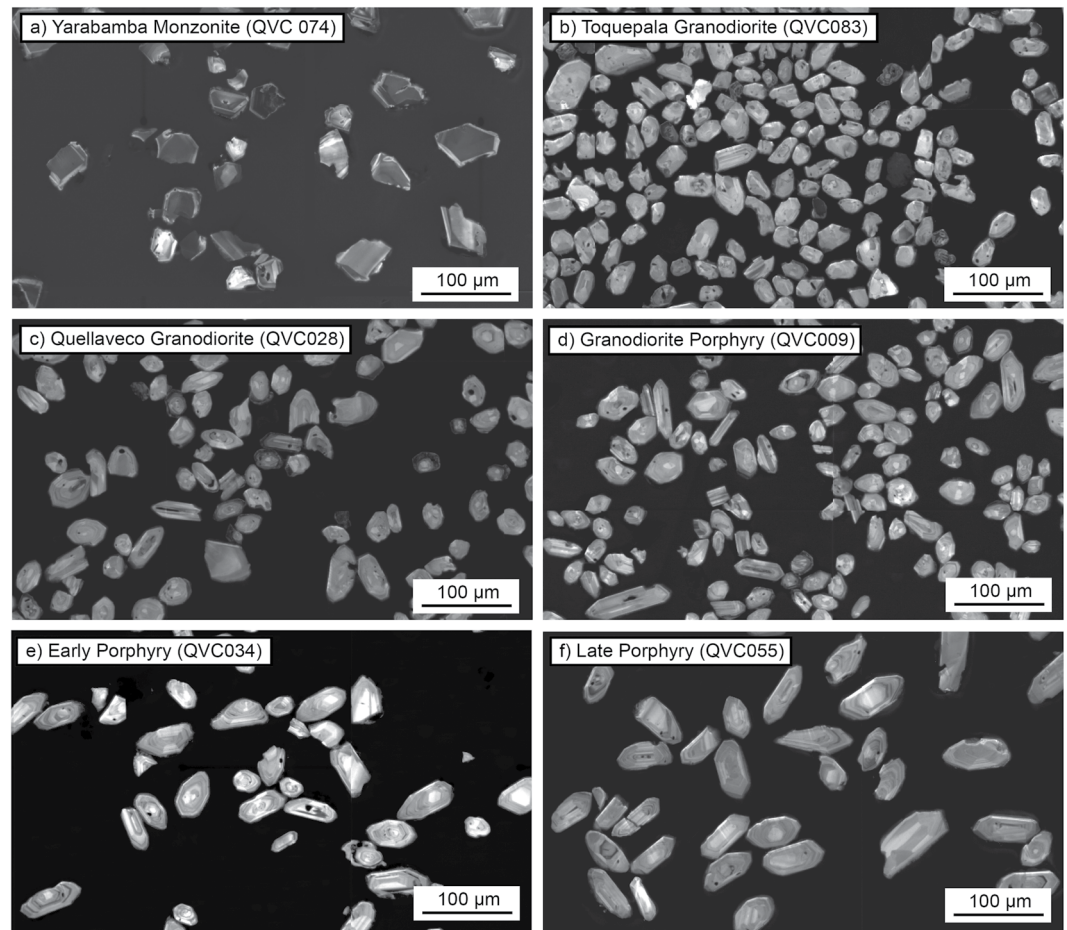
Figure 4. SEM-CL images of zircon mounts from six representative samples showing the range of morphologies, sizes, and textures of zircons within the studied samples. The scale is identical for all six images.

3. Results

3.1. Zircon Textures

The zircons studied show a range of morphologies, sizes, and textures which vary between the different lithologies (Figures 4 and 5 and Table 1). Zircons from the Yarabamba Monzonite, particularly sample QVC074, show a distinct morphology and texture (Figure 4a). These zircons are anhedral with irregular edges and often contain fractures. The majority of crystals in this unit are generally unzoned, large (>50 μm) crystals with a weak CL response surrounded by a bright, narrow rim. A smaller number of crystals in this unit are smaller (>50 μm), with a bright CL response, strong oscillatory zoning, and large amounts of mineral/melt inclusions. The Toquepala Granodiorite sample contains subhedral, small (>50 μm) zircons which are variably oscillatory zoned (Table 1 and Figure 5). Zircons from the two Quellaveco Granodiorite samples (QVC028 and QVC058), the youngest member of the Yarbamba Batholith, are indistinguishable from the Quellaveco porphyry zircons (Figure 4c). These porphyry zircons show euhedral morphologies characterized by pyramidal and prismatic forms, are generally large (>70 μm and often >100 μm) and are strongly oscillatory and/or weakly sector zoned (Figures 4d–4f). A proportion of these zircons (~20%) have small, unzoned cores surrounded by a strongly oscillatory zoned exteriors. In all units, a small to moderate proportion of zircon crystals contain inclusions which are predominantly apatite and melt inclusions.

Based on the association of rocks with mineralization, two populations of zircon were defined. The first population, termed "porphyry," includes all zircons from the Quellaveco porphyry rocks and the host granodiorite. The Quellaveco Granodiorite (the latest member of the Yarabamba Batholith) was included in this population since it displays geochemistry (both whole rock and zircon) akin to the porphyry intrusions and is thought to be
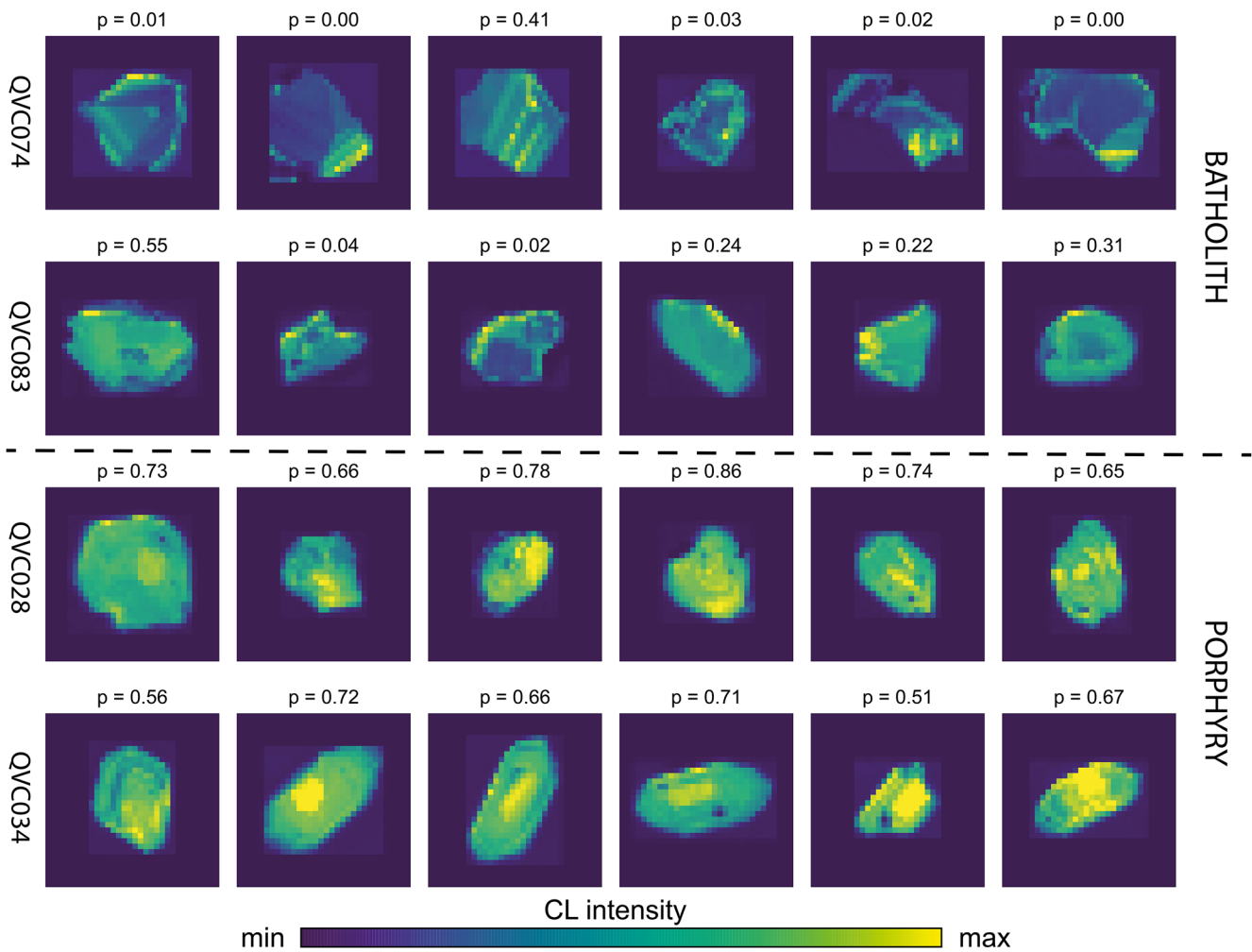
Figure 5. Randomly selected images of zircons from four samples. QVC074 (Yarabamba Monzonite)—zircons are anhedral with dull, unzoned cores, and bright rims. QVC083 (Toquepala Granodiorite)—zircons are subhedral with more variable textures but often weakly zoned. QVC028 (Quellaveco Granodiorite) and QVC034 (Early Porphyry)—zircons are euhedral with oscillatory zoning. For each image, the trained LeNet model output (p) is given above, which indicates the probability that the zircon belongs to the "porphyry" population as determined by the CNN. Note that these zircons are within the data set used to train the model. Images are converted from grayscale to *viridis* color scale to increase contrast.

associated with minor mineralization in the district (Nathwani et al., 2021; Simmons, 2013). In addition, initial textural examination of the crystals indicates that the Quellaveco Granodiorite exhibits textures identical to the porphyry intrusions (comparing Figures 4c–4f). Therefore labeling these zircon images as porphyry-related will lead to a more robust classification model. The second population, termed "batholith," includes the two intrusive phases of the Yarabamba Batholith which significantly predate mineralization (the Yarabamba Monzonite and Toquepala Granodiorite).

A total of 750 images are available in this data set, with 72 of these images (from two samples) withheld in the independent test set. Six hundred and seventy-eight images comprised the training and validation set and the "porphyry" and "batholith" populations yielded 409 and 269 images, respectively. These images were randomly split into training and validation subsets using an 80:20 ratio in order to test model performance after each training epoch.

3.2. Model Training and Validation

The accuracy and ROC-AUC of the training and validation data set generally increases after each training epoch, while the loss decreases (Figure 6). For the LeNet-5 and AlexNet models, with a smaller number of training

Table 1
Summary of Samples Used in the Train and Test Data Sets for the Zircon CNN Models From the Quellaveco Porphyry Cu District

Sample	Lithology	Age (Ma)	Train/test	Zircon notes	Classification	n
QVC074	Yarabamba Monzonite	–	Train	Anhedral, dark unzoned cores, bright rims	Batholith	175
QVC075	Yarabamba Monzonite	67.18 ± 1.56	Test	Anhedral-euhedral, variable zoning	Batholith	46
QVC082	Toquepala Granodiorite	60.96 ± 1.39	Train	Euhedral, oscillatory zoning	Batholith	21
QVC083	Toquepala Granodiorite	61.64 ± 1.26	Train	Euhedral, variable zoning	Batholith	122
QVC028	Quellaveco Granodiorite	58.99 ± 1.20	Train	Euhedral, strong oscillatory zoning	Porphyry	103
QVC058	Quellaveco Granodiorite	–	Train	Euhedral, strong oscillatory zoning	Porphyry	43
QVC009	Granodiorite Porphyry	–	Train	Euhedral, strong oscillatory zoning	Porphyry	62
QVC034	Early Porphyry	54.40 ± 1.11	Train	Euhedral, strong oscillatory zoning	Porphyry	49
QVC024	Intermineral Porphyry	54.19 ± 1.10	Train	Euhedral, strong oscillatory zoning	Porphyry	64
QVC060	Intermineral Porphyry	54.88 ± 1.12	Train	Euhedral, strong oscillatory zoning	Porphyry	40
QVC061	Monzodiorite Porphyry	55.26 ± 1.15	Train	Euhedral, strong oscillatory zoning	Porphyry	30
QVC055	Late Porphyry	55.45 ± 1.16	Test	Euhedral, strong oscillatory zoning	Porphyry	25
QVC029	Late Porphyry	54.68 ± 1.12	Train	Euhedral, strong oscillatory zoning	Porphyry	41

Note. Ages are emplacement ages derived from the weighted mean of LA ICP-MS U-Pb zircon dates from Nathwani et al. (2021). Brief notes on the key textures of zircons in each sample are given plus the class label and the number of images used in the train or test sets.

epochs (<100), the validation set performance generally exceeds the training set and the performances converge as the training process reaches 200 epochs (Figure 6 and Table 2). In cases such as this, where the validation performance exceeds the training performance, poor generalization is indicated, perhaps due to the small size of the data set and potentially due to the fact that dropout was turned off during the validation process so all neurones in the network were available (Srivastava et al., 2014). The CNN models greatly outperform the baseline, where the baseline is the ANN without any convolutional or pooling layers (Figure 6).

Models with differing components of regularization show large differences in performance metrics on the validation data set (Figure S2 in Supporting Information S1 and Table 2). With no regularization, the performance (particularly the loss and ROC-AUC) decreases after 20 training epochs, indicating that the model has overfit the training data (Figure S2 in Supporting Information S1). Adding dropout regularization (with a probability of 0.5) generally appears to mitigate against overfitting, and improves model performance. Adding data augmentation also appears to improve model performance and, after 200 epochs of training, there is no evidence for overfitting. Combining dropout and data augmentation produces a similar to marginally better performance. This
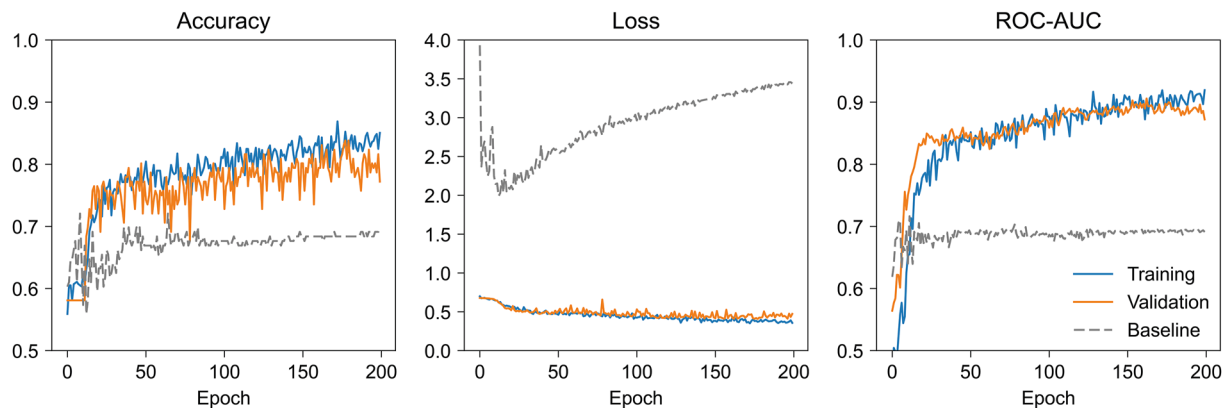
Figure 6. Plots showing changes in accuracy, loss, and receiver operating characteristic-area under curve (ROC-AUC) during the training process. Metrics after each training epoch are shown, which is one complete pass of the training data set through the model. The similar metrics for both the training (blue) and validation (orange) suggests that the model has generalized the data well. The gray dashed line (baseline) indicates the validation data set performance of a fully connected artificial neural network (ANN).

Table 2
Summary of Model Architectures Implemented in This Study and the Performance Metrics for Training, Validation, and Testing of the Models

Model		LeNet-5	LeNet-5	LeNet-5	LeNet-5	AlexNet	VGG
Regularization		None	Dropout	Data augmentation	Dropout + data augmentation	Dropout + data augmentation + BatchNorm	Dropout + data augmentation
Parameters		22,029	22,029	22,029	22,029	10,347,009 (2,752)	14,784,513
Training		From scratch	From scratch	From scratch	From scratch	From scratch	Transfer learning
Train	Loss	0.00	0.04	0.29	0.41	0.25	0.26
	Accuracy	1.00	0.99	0.88	0.81	0.91	0.89
	ROC-AUC	1.00	1.00	0.95	0.89	0.96	0.96
Validate	Loss	0.96	0.73	0.30	0.31	0.31	0.24
	Accuracy	0.82	0.81	0.83	0.88	0.86	0.88
	ROC-AUC	0.89	0.91	0.95	0.94	0.95	0.96
Test	Loss	2.90	0.43	1.36	0.59	1.09	0.40
	Accuracy	0.69	0.83	0.69	0.75	0.56	0.86
	ROC-AUC	0.81	0.91	0.74	0.88	0.79	0.93

Note. Parameters in parentheses indicate those which are nontrainable (from batch normalization).

best performing LeNet model achieves an accuracy close to 0.8 and an ROC-AUC of 0.9, indicating that the model has a 90% probability of ranking a randomly selected "porphyry" zircon higher than a "batholith" zircon. The validity of these metrics based on a training-validation split was further tested by performing a fivefold cross-validation, where 80:20 splits were performed five times until all data appeared in the validation data set once. This yielded comparable performance metrics: an accuracy of 0.81 ± 0.02, a loss of 0.43 ± 0.06, and an ROC-AUC of 0.90 ± 0.03 (Table S2 in Supporting Information S1), suggesting the training-validation split and metrics are valid. Overall, the performance suggests that CNNs can effectively discriminate the two populations, and that employing regularization is paramount in producing models that generalize well.

The AlexNet model shows greater variability during the training process (Figure 7), likely due to the fact it contains 2 orders of magnitude more trainable parameters, which is substantial relative to the number of training examples and the mini-batch size. After 200 epochs, the model achieves a similar classification performance to the regularized LeNet-style models, with accuracy and ROC-AUC on the validation data set of 0.86 and 0.95, respectively (Table 2).

The transfer learning approach with the VGG-16 model generally shows the best performance, with accuracy and ROC-AUC scores of 0.88 and 0.96 on the validation set. This improved performance relative to other models is
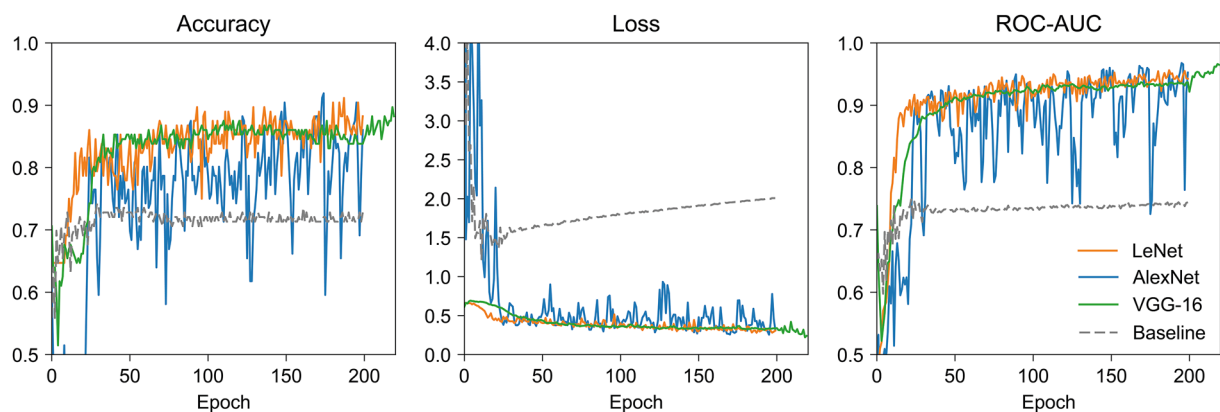
Figure 7. A comparison of the performance of each model on the validation set during the training process. The baseline is the multilayer perceptron described in the text (i.e., with no convolutional layers or pooling layers). Note that the VGG16 model is trained for 20 more epochs at a lower training rate (0.0001) in which the final convolutional layer was unfrozen, which was found to improve model performance.
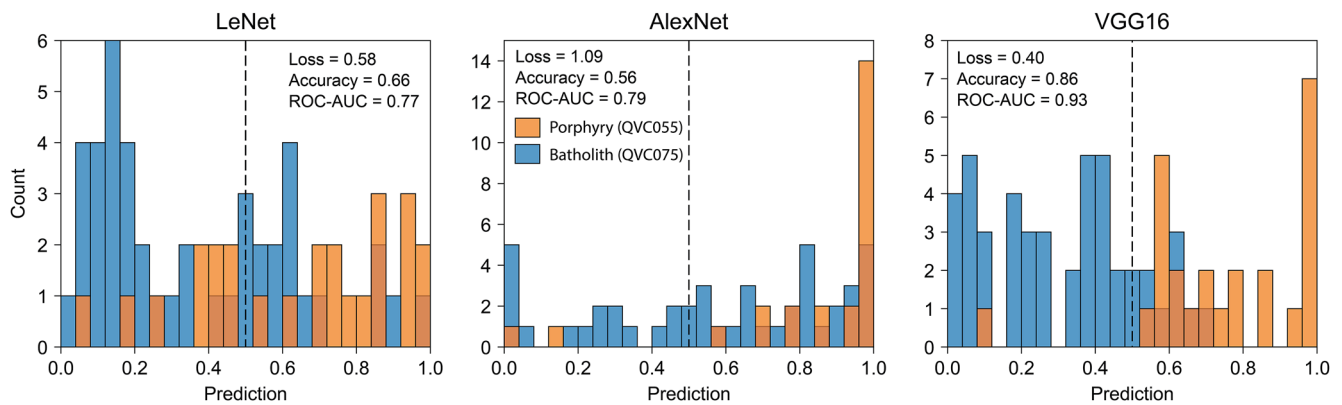
Figure 8. Histograms showing the convolutional neural network (CNN) output probabilities for the two samples in the test data set, from the LeNet (includes dropout and data augmentation), AlexNet and VGG16 models. The loss, accuracy, and receiver operating characteristic-area under curve (ROC-AUC) for each model test are shown.

largely caused by the "fine tuning" stage in which the model is trained for 20 further epochs at a lower learning rate with the final convolutional layer unfrozen.

### 3.3. Hyperparameter Selection

Building CNN models requires manual selection of a number of parameters that are not learned during the training process, known as hyperparameters. Typically, the optimal parameters will be selected by running several models with varying parameters and selecting the best model. In general, the size, stride, and number of convolutional layers were selected in order to progressively, and evenly, decrease the image size, while simultaneously increasing the number of channels, because this has been shown to be optimal for CNNs (Krizhevsky et al., 2012; LeCun & Bengio, 1998). The models were adjusted and retrained with higher resolution images, of 64×64 pixels rather than 32×32 pixels. These images contain larger amounts of information, such as a more detailed record of zoning. However, using higher resolution images was found to not produce a discernible increase in model performance. Furthermore, the LeNet model with a 64×64 input contains 4.7 times as many trainable parameters (156,645 compared to 33,765) and was thus more computationally expensive to run and also more prone to overfitting. In conclusion, using the lower resolution images was found not to result in a loss of useful information for classification purposes. Other hyperparameters that were tested were: the learning rate, the optimization algorithm, the activation function, the effect of padding during convolutions, the stride of convolutions, and the kernel size. The full list of hyperparameters that were tested can be found in Table S5 in Supporting Information S1.

3.4. Model Testing

The trained LeNet model (with data augmentation and dropout) also performed well on the two independent test samples from the Yarabamba Monzonite ("batholith") and the Late Quellaveco Porphyry ("porphyry") showing comparable metrics to the validation set (Table 2). Examining the distributions of probabilities produced for each sample clearly shows that the bulk population lies greater than and less than the classification threshold for the "porphyry" and "batholith" rock, respectively (Figure 8). The trained AlexNet model performed slightly less well on the test set relative to its validation performance (accuracy = 0.56 and ROC-AUC = 0.79; Table 2 and Figure 8). This poorer generalization is likely the result of the large number of model parameters relative to the number of images in the training data set. The transfer learning VGG-16 approach performed best on the test data set, producing comparable performance metrics relative to validation performance (accuracy = 0.86 and ROC-AUC = 0.93; Table 2 and Figure 8). Therefore, the transfer learning approach appears to produce the best model generalization. The ability of a CNN model to better generalize when pretrained on the ImageNet data set highlights the ability for transfer learning to produce exceptional results even in the absence of a large training data set. This finding is in agreement with work in other fields such as remote sensing (Pires de Lima & Marfurt, 2020) and prospectivity mapping (Li et al., 2020; Wu et al., 2022).

In order to identify possible features of importance, it is possible to individually examine crystals in the data sets, to compare the true class label with the predicted label and its probability. Generally, the trained LeNet model
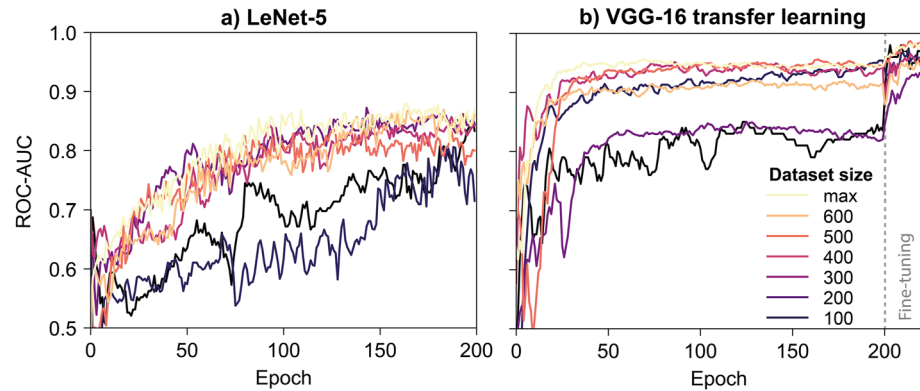
Figure 9. Comparison of receiver operating characteristic-area under curve (ROC-AUC) scores for (a) LeNet-5 and (b) VGG-16 transfer learning approaches, as the size of the data set is changed. The maximum size (max) is 678 images. The ROC-AUC for the validation set is shown. Note that the data set size is the sum of the training and validation sets. The vertical dashed line in b indicates when fine tuning of the model begins (for 20 epochs; see text for further details).

confidently predicts an image as "porphyry" if the crystal is euhedral and exhibits oscillatory zoning (Figure 5). On the contrary, if a zircon crystal is anhedral and weakly zoned, the model predicts the image as "batholith" with high confidence. For more complex crystals, the model has lower confidence in its prediction. As an example, a strongly oscillatory zoned image that represents a fragment of a crystal, rather than an entire crystal, is misclassified (with low confidence), likely because the model is not familiar with this morphology.

3.5. Data Set Size and Transfer Learning

Relative to other studies implementing deep CNNs to solve computer vision tasks, the data set size here is relatively small. For example, it has been proposed that the number of training examples should be 10 times the number of parameters in a deep learning model (Miotto et al., 2018). The data set size here (678 images) is orders of magnitude less than the number of parameters in some CNN models (e.g., over 10 million for AlexNet). Therefore, training such large models is prone to overfitting and poor generalization (Nasir & Sassani, 2021), and may contribute to the strong performance of the transfer learning approach where only a fraction of parameters are adjusted. The implementation of data augmentation also likely plays a key role in improving generalization in the models trained here.

In order to evaluate the role of data set size on performance, the LeNet and VGG (transfer learning) models were retrained with progressively smaller, randomly subsampled data sets (Figure 9a and Figures S3, S4 in Supporting Information S1). Decreasing the data set size causes a progressive decrease in performance of the LeNet model, and leads to high variance of performance metrics during the training process (Figure 9a). With a data set greater than 400 images, the performance only shows marginal performance gains, suggesting data set size is a less strong limiting factor in model performance beyond this. The surprisingly high performance with only 400 images, may be the result of the relatively low complexity of the crystal features that the models must recognize in order to make correct predictions (e.g., crystal and zoning morphology). The transfer learning approach with the VGG-16 model (Figure 9b) also shows poorer performance with decreasing data set size. As discussed previously, the greater performance of this approach, relative to training the LeNet-5 model from scratch is evident here. Notably, this approach, regardless of the number of training examples, is able to achieve stronger performance after a short training time (50 epochs), demonstrating that the transfer learning approach can be used as a computationally cheaper methodology. Overall, with only a data set of 400 zircon CL images, our models perform well. A transfer learning approach serves as the best approach to overcome the challenges of small data sets, which may be particularly common in geoscience studies. Furthermore, our trained VGG-16 model may provide a useful weight initialization for future studies applying CNNs to classify crystal textures.

3.6. Feature Extraction

The strong performance of the LeNet model was assessed carefully to ensure that image discrimination was based on texture and/or morphology, rather than on undesirable features, such as contrast. One useful method for
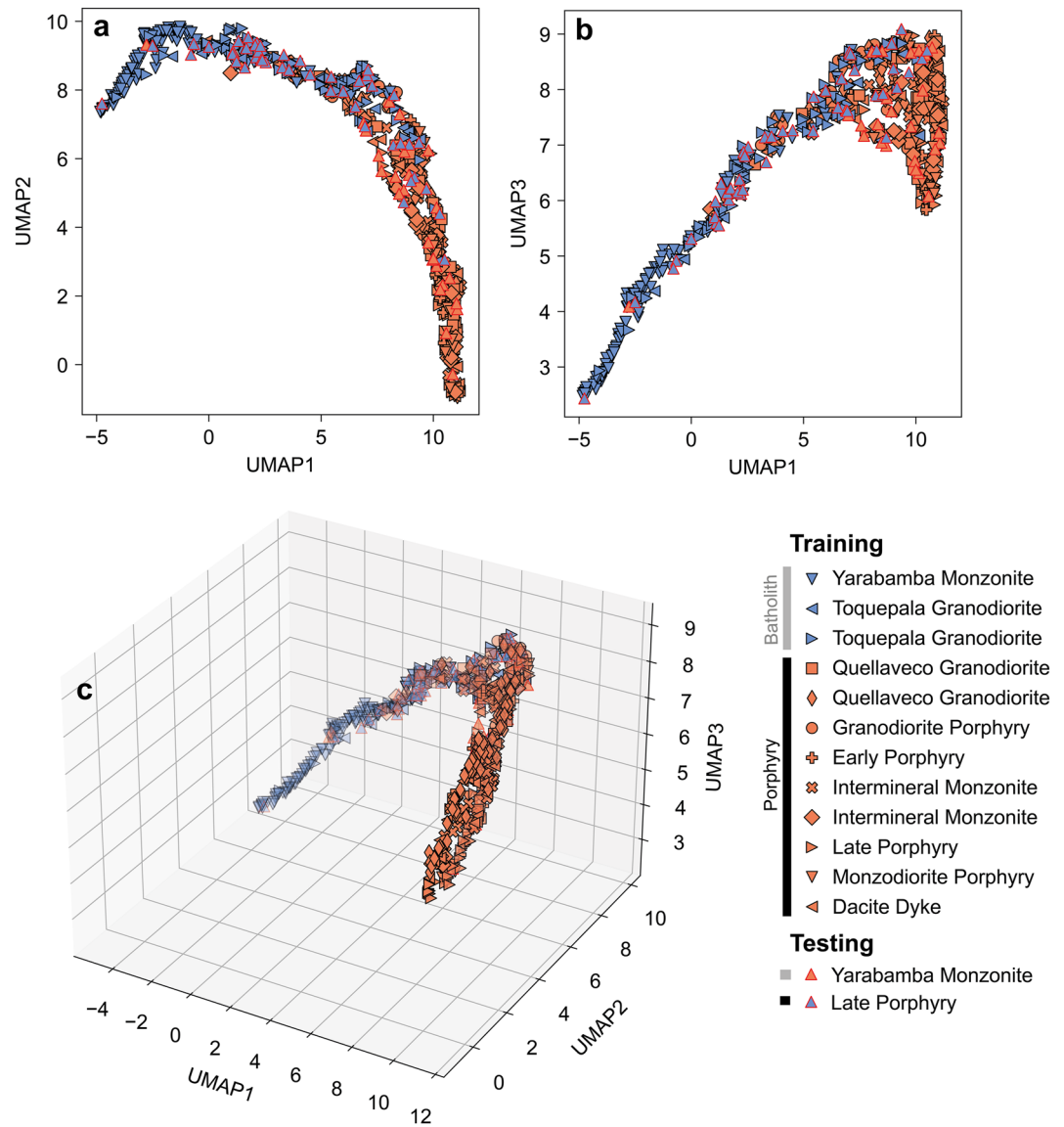
Figure 10. Dimensionality reduction of the LeNet model's penultimate layer output (32-dimensional vector) for the training and test data set. The clustering algorithm used is the UMAP algorithm, which reduces the 32 dimensions into 3 dimensions (UMAP1, UMAP2, and UMAP3). (a) and (b) show 2D projections of UMAP embedding vectors and (c) shows 3D projection.

evaluating the behavior of CNN models is to remove the final layer and extract results from the penultimate (fully connected) layer (Chen et al., 2016). In this instance, this approach will produce a 32-dimensional vector for each training example. The 32-dimensional outputs for all training examples then can be clustered using a dimensionality reduction algorithm; in this case, we implemented the UMAP algorithm (Figure 10), which uses manifold learning techniques (McInnes et al., 2020). The UMAP algorithm generates a probability distribution between high dimensional data points, where similar data are assigned a high probability and less similar data are assigned a low probability. These are then compared to probabilities in a low dimension space (e.g., 2D or 3D) and the algorithm minimizes the difference (measured by the cross-entropy cost function) between the two probability distributions. This dimensionality reduction technique has an advantage over principal component analysis in that it can capture nonlinear relationships. It is also commonly preferred to its predecessor, the t-distributed stochastic neighbor embedding (t-SNE) algorithm, due to its speed and better ability to preserve global data structure (McInnes et al., 2020). The UMAP algorithm was implemented using the Python *umap-learn* package (v. 0.5.1; Sainburg et al., 2021).
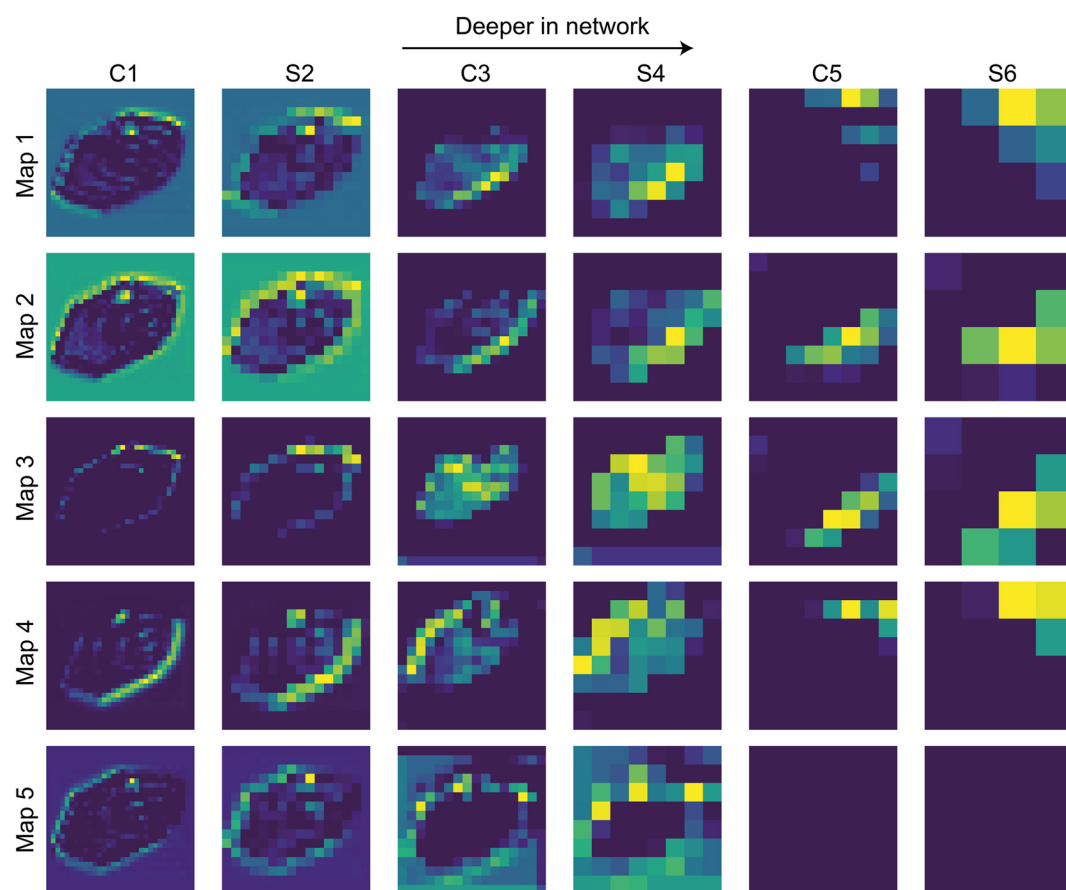
Figure 11. Feature maps extracted from the convolutional neural network (CNN) model for a zircon in the training data set. Each columns show the first five feature maps for each layer, with maps from deeper in the network from left to right. The labeled columns indicate the abbreviated layer name as shown in Figure 2. In the first layer (C1), the feature maps extract different edges from the image, such as particular zoning patterns, crystal morphology, and an apatite inclusion in the top portion of the crystal. The max pooling layers (e.g., S2) reduce the dimensionality of the image while capturing the key information. Moving deeper in the network, the feature maps become progressively less interpretable.

The feature extraction results clearly show that samples QVC074 and QVC075 (both "batholith") are the most distinct from all other samples, characterized by a predominance of diagnostic unzoned and anhedral zircon crystals. On the contrary, samples QVC082 and QVC083 (both also "batholith") are less distinct, yet partly distinguishable, from the porphyry zircons; this is consistent with their subhedral shapes and weakly zoned textures. The "porphyry" zircons partly overlap and are partly distinct from the batholith zircons, in UMAP space, reflecting their tendency to be dominated by strongly zoned and euhedral zircons. As noted earlier, the Quellaveco Granodiorite zircons, which are derived from an intrusion which is transitional between batholith construction and porphyry formation, do not show any clear difference from the Quellaveco porphyry zircons. Another interesting observation is that the UMAP dimensions generally correspond to particular features of a zircon image where UMAP1, UMAP2, and UMAP3 are interpreted to reflect the crystal shape, frequency/abundance of zoning, and the intensity of CL response, respectively. Overall, this feature extraction approach corroborates observational differences between zircons from each sample, and the ability of the model to discriminate "porphyry" and "batholith" crystals based on real textural and morphological differences, thus providing important model understandability.

Another useful approach for evaluating CNN behavior is to extract the results from individual convolutional layers for a given training example (Chollet, 2017). For a randomly selected zircon, the output of intermediate layers in the LeNet-5 model was extracted and a subset of these are presented (Figure 11 shows 5 maps for the first six layers). Two more examples can be found in the Supporting Information S1 (Figures S5 and S6). It is evident that the first convolutional layer (C1) is extracts the larger scale features of a zircon crystal, such as the

crystal edges and the zoning patterns. In addition, the model is clearly able to recognize the presence of an apatite inclusion in the uppermost portion of the crystal. Pooling layers (e.g., S2) reduce the size of the feature extracted maps while preserving key features. On moving deeper in the network, the activations become increasingly more abstract and difficult to interpret. These layers are mostly encoding higher level features such as angles or corners.

4. Discussion

4.1. Petrogenetic Implications

The predominance of strongly oscillatory zoned zircons in the Quellaveco porphyry intrusions supports previous observations from other porphyry Cu systems (Chelle-Michou et al., 2014; Leslie et al., 2021; Lu et al., 2016; Pizarro et al., 2020). In addition, the occurrence of unzoned cores and strongly oscillatory zoned exteriors in some Quellaveco porphyry zircons is in accordance with studies of zircons in other porphyry Cu deposits (Lu et al., 2016). Similarly, more variable zircon textures (weakly/patchy zoned or homogeneous) in premineralization batholiths which are not associated with porphyry Cu mineralization in the district have also been described previously, such as at Coroccohuayco, Peru (Chelle-Michou et al., 2014).

Although oscillatory zoned textures and euhedral crystal shapes appear to be particularly common in porphyry zircons (Aghazadeh et al., 2015; Buret et al., 2016; Chelle-Michou et al., 2014; Large et al., 2018, 2020; Lee et al., 2017; Loader et al., 2022), oscillatory zoning can be present in zircons from a variety of other granitoids (Corfu et al., 2003; Fowler et al., 2002; Hoskin, 2000). Generally, brighter oscillatory zones are characterized by higher concentrations of U, Th, and REEs (Fowler et al., 2002). Hoskin (2000) suggested that oscillatory zoning in zircon is driven by dynamics at the zircon/melt interface involving cation substitution, diffusion, and melt polymerization. Chemical heterogeneities are generated by the fast growth rate of a zircon crystal relative to the diffusion of high field strength element ions, reflecting kinetic factors rather than changes in mineral-melt partitioning (Hofmann et al., 2014). Recent modeling by Melnik and Bindeman (2018) showed that fluctuations in temperature and pressure during recharge and convection within magmatic systems can explain oscillatory zoning in zircon. They further added that oscillatory zoning in zircon is most readily produced in wet, crystal-rich magmas with temperatures <700°C. The oscillatory zoned nature of the Quellaveco porphyry zircons could therefore reflect the more evolved, cooler, fluid-saturated compositions of the Quellaveco magmas, compared to the hotter, intermediate, and the lower water contents inferred for the Yarabamba magmas (Nathwani et al., 2021).

Regardless of the origin, the ability to rapidly discriminate textural populations using CNNs may provide a novel and effective exploration approach. The outputs derived from such models could be used as an initial screening process and subsequently integrated with zircon trace element compositions (Loader et al., 2017; Lu et al., 2016; Zou et al., 2022) in assessments of porphyry copper potential, and would be especially effective for analysis of zircon concentrates separated from stream sediments.

4.2. Future Work

This work highlights the potential for automated textural recognition and the classification of zircons from porphyry Cu districts. Further work should also be done to train such models on a more diverse range of lithologies that may be encountered in a porphyry district such as Quellaveco (e.g., more unmineralized intrusions and volcanic rocks). Testing of such models on detrital zircons in stream sediments should be done to test the potential of the approach for identifying porphyry zircons in catchment areas. These applications to detrital zircons may require additional work because further training of the models on detrital zircons may be required to yield model outputs invariant to loss of shape due to sedimentary transport. The approach can also be applied to other districts, and eventually can be applied on a broader scale to previously unexplored areas. If successful, such tools could be used, alongside whole rock geochemistry, zircon trace element chemistry, and geochronology, to assist in providing an assessment of porphyry Cu fertility. CNN classifiers could also be employed for studies on other indicator minerals, such as apatites, which show textural variability (Bouzari et al., 2016). The approach could also be applied to other petrological tasks, such as distinguishing crystal cargo populations erupted at volcanoes (Sheldrake & Higgins, 2021) and integrating this with geochemistry could help better constrain the evolution of magmatic plumbing systems. Further development of unsupervised CNN approaches also has significant potential to discriminate and cluster crystal populations for these purposes. Such methodologies are also applicable in sedimentology, where textures of grains can be used to constrain sedimentary provenance (Moral Cardona et al., 2005).

5. Conclusions

This study has shown that CNNs provide a viable method for automated mineral texture recognition and discrimination. The method was capable of distinguishing zircons from intrusions associated with Quellaveco porphyry Cu deposit, from zircons from the premineralization Yarabamba Batholith with reasonable accuracy. The results indicate that the porphyry zircons can be ranked at a higher probability of belonging to a "porphyry" rock than a batholith-derived zircon with an ~85% success rate. This represents a novel approach to enable rapid discrimination of crystal populations in mineral exploration and provenance studies.

Data Availability Statement

The code and data for this study are available at https://github.com/ChetanNathwani/zirconCNN and archived at the point of submission at https://doi.org/10.5281/zenodo.7260821.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., et al. (2016). TensorFlow: A system for large-scale machine learning (pp. 265–283). Retrieved from https://www.usenix.org/conference/osdi16/technical-sessions/presentation/abadi

Aghazadeh, M., Hou, Z., Badrzadeh, Z., & Zhou, L. (2015). Temporal-spatial distribution and tectonic setting of porphyry copper deposits in Iran: Constraints from zircon U-Pb and molybdenite Re-Os geochronology. *Ore Geology Reviews*, *70*, 385–406. https://doi.org/10.1016/j.oregeorev.2015.03.003

Ballard, J. R., Palin, M. J., & Campbell, I. H. (2002). Relative oxidation states of magmas inferred from Ce(IV)/Ce(III) in zircon: Application to porphyry copper deposits of northern Chile. *Contributions to Mineralogy and Petrology*, *144*(3), 347–364. https://doi.org/10.1007/s00410-002-0402-5

Bouzari, F., Hart, C. J., Bissig, T., & Barker, S. (2016). Hydrothermal alteration revealed by apatite luminescence and chemistry: A potential indicator mineral for exploring covered porphyry copper deposits*. *Economic Geology*, *111*(6), 1397–1410. https://doi.org/10.2113/econgeo.111.6.1397

Bradski, G. (2000). *The OpenCV Library. Dr. Dobb's Journal: Software Tools for the Professional Programmer*, *25*(11). Retrieved from https://www.elibrary.ru/item.asp?id=4934581

Buret, Y., von Quadt, A., Heinrich, C., Selby, D., Wälle, M., & Peytcheva, I. (2016). From a long-lived upper-crustal magma chamber to rapid porphyry copper emplacement: Reading the geochemistry of zircon crystals at Bajo de la Alumbrera (NW Argentina). *Earth and Planetary Science Letters*, *450*, 120–131. https://doi.org/10.1016/j.epsl.2016.06.017

Canny, J. (1986). A computational approach to edge detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-8, (6), 679–698. https://doi.org/10.1109/TPAMI.1986.4767851

Chelle-Michou, C., Chiaradia, M., Ovtcharova, M., Ulianov, A., & Wotzlaw, J.-F. (2014). Zircon petrochronology reveals the temporal link between porphyry systems and the magmatic evolution of their hidden plutonic roots (the Eocene Coroccohuayco deposit, Peru). *Lithos*, *198–199*, 129–140. https://doi.org/10.1016/j.lithos.2014.03.017

Chen, Y., Jiang, H., Li, C., Jia, X., & Ghamisi, P. (2016). Deep feature extraction and classification of hyperspectral images based on convolutional neural networks. *IEEE Transactions on Geoscience and Remote Sensing*, *54*(10), 6232–6251. https://doi.org/10.1109/TGRS.2016.2584107

Chollet, F. (2015). keras. Retrieved from https://github.com/keras-team/keras

Chollet, F. (2017). *Deep learning with Python*. Simon and Schuster.

Cooke, D. R., Agnew, P., Hollings, P., Baker, M., Chang, Z., Wilkinson, J. J., et al. (2020). Recent advances in the application of mineral chemistry to exploration for porphyry copper-gold-molybdenum deposits: Detecting the geochemical fingerprints and footprints of hypogene mineralization and alteration. *Geochemistry: Exploration, Environment, Analysis*, *20*(2), 176–188. https://doi.org/10.1144/geochem2019-039

Corfu, F., Hanchar, J. M., Hoskin, P. W., & Kinny, P. (2003). Atlas of zircon textures. *Reviews in Mineralogy and Geochemistry*, *53*(1), 469–500. https://doi.org/10.2113/0530469

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition* (pp. 248–255). IEEE. https://doi.org/10.1109/CVPR.2009.5206848

Dilles, J. H., Kent, A. J., Wooden, J. L., Tosdal, R. M., Koleszar, A., Lee, R. G., & Farmer, L. P. (2015). Zircon compositional evidence for sulfur-degassing from ore-forming arc magmas. *Economic Geology*, *110*(1), 241–251. https://doi.org/10.2113/econgeo.110.1.241

Dramsch, J. S. (2020). Chapter One—70 years of machine learning in geoscience in review. In B. Moseley, & L. Krischer (Eds.), *Advances in geophysics* (Vol. 61, pp. 1–55). Elsevier. https://doi.org/10.1016/bs.agph.2020.08.002

Fowler, A., Prokoph, A., Stern, R., & Dupuis, C. (2002). Organization of oscillatory zoning in zircon: Analysis, scaling, geochemistry, and model of a zircon from Kipawa, Quebec, Canada. *Geochimica et Cosmochimica Acta*, *66*(2), 311–328. https://doi.org/10.1016/s0016-7037(01)00774-8

Glorot, X., Bordes, A., & Bengio, Y. (2011). Deep sparse rectifier neural networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics* (pp. 315–323). Retrieved from https://proceedings.mlr.press/v15/glorot11a.html

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. arXiv:1207.0580.

Hofmann, A. E., Baker, M. B., & Eiler, J. M. (2014). Sub-micron-scale trace-element distributions in natural zircons of known provenance: Implications for Ti-in-zircon thermometry. *Contributions to Mineralogy and Petrology*, *168*(3), 1057. https://doi.org/10.1007/s00410-014-1057-8

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, *2*(5), 359–366. https://doi.org/10.1016/0893-6080(89)90020-8

Hoskin, P. W. O. (2000). Patterns of chaos: Fractal statistics and the oscillatory chemistry of zircon. *Geochimica et Cosmochimica Acta*, *64*(11), 1905–1923. https://doi.org/10.1016/s0016-7037(00)00330-6

Hrstka, T., Gottlieb, P., Skala, R., Breiter, K., & Motl, D. (2018). Automated mineralogy and petrology—Applications of TESCAN Integrated Mineral Analyzer (TIMA). *Journal of Geosciences*, *63*(1), 47–63. https://doi.org/10.3190/jgeosci.250

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning* (pp. 448–456). PMLR. Retrieved from http://proceedings.mlr.press/v37/ioffe15.html

Kingma, D. P., & Ba, J. (2017). Adam: A method for stochastic optimization. arXiv:1412.6980

Koeshidayatullah, A., Morsilli, M., Lehrmann, D. J., Al-Ramadan, K., & Payne, J. L. (2020). Fully automated carbonate petrography using deep convolutional neural networks. *Marine and Petroleum Geology*, *122*, 104687. https://doi.org/10.1016/j.marpetgeo.2020.104687

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems* (Vol. 1, pp. 1097–1105). Red Hook, NY: Curran Associates Inc.

Large, S. J. E., Quadt, A. V., Wotzlaw, J.-F., Guillong, M., & Heinrich, C. A. (2018). Magma evolution leading to porphyry Au-Cu mineralization at the Ok Tedi deposit, Papua New Guinea: Trace element geochemistry and high-precision geochronology of igneous zircon. *Economic Geology*, *113*(1), 39–61. https://doi.org/10.5382/econgeo.2018.4543

Large, S. J. E., Wotzlaw, J.-F., Guillong, M., von Quadt, A., & Heinrich, C. A. (2020). Resolving the timescales of magmatic and hydrothermal processes associated with porphyry deposit formation using zircon U-Pb petrochronology. *Geochronology*, *2*(2), 209–230. https://doi.org/10.5194/gchron-2-209-2020

LeCun, Y., & Bengio, Y. (1998). Convolutional networks for images, speech, and time series. In *The handbook of brain theory and neural networks* (Vol. 3361(10), pp. 255–258). ACM.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*, 436–444. https://doi.org/10.1038/nature14539

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation*, *1*(4), 541–551. https://doi.org/10.1162/neco.1989.1.4.541

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278–2324. https://doi.org/10.1109/5.726791

Lee, R. G., Dilles, J. H., Tosdal, R. M., Wooden, J. L., & Mazdab, F. K. (2017). Magmatic evolution of granodiorite intrusions at the El Salvador porphyry copper deposit, Chile, based on trace element composition and U/Pb age of zircons‡. *Economic Geology*, *112*(2), 245–273. https://doi.org/10.2113/econgeo.112.2.245

Leslie, C., Meffre, S., Cooke, D., & Thompson, J. (2021). Complex petrogenesis of porphyry-related magmas in the Cowal District, Australia: Insights from LA ICP-MS zircon imaging. *SEG Special Publications*, *24*(2), 159–180.

Li, H., Li, X., Yuan, F., Jowitt, S. M., Zhang, M., Zhou, J., et al. (2020). Convolutional neural network and transfer learning based mineral prospectivity modeling for geochemical exploration of Au mineralization within the Guandian-Zhangbaling area, Anhui Province, China. *Applied Geochemistry*, *122*, 104747. https://doi.org/10.1016/j.apgeochem.2020.104747

Li, T., Zuo, R., Xiong, Y., & Peng, Y. (2021). Random-drop data augmentation of deep convolutional neural network for mineral prospectivity mapping. *Natural Resources Research*, *30*(1), 27–38. https://doi.org/10.1007/s11053-020-09742-z

Loader, M. A., Nathwani, C. L., Wilkinson, J. J., & Armstrong, R. N. (2022). Controls on the magnitude of Ce anomalies in zircon. *Geochimica et Cosmochimica Acta*, *328*, 242–257. https://doi.org/10.1016/j.gca.2022.03.024

Loader, M. A., Wilkinson, J. J., & Armstrong, R. N. (2017). The effect of titanite crystallisation on Eu and Ce anomalies in zircon and its implications for the assessment of porphyry Cu deposit fertility. *Earth and Planetary Science Letters*, *472*, 107–119. https://doi.org/10.1016/j.epsl.2017.05.010

Lu, Y.-J., Loucks, R. R., Fiorentini, M., McCuaig, T. C., Evans, N. J., Yang, Z.-M., et al. (2016). Zircon compositions as a pathfinder for porphyry Cu ± Mo ± Au deposits. *SEG Special Publications: Tectonics and Metallogeny of the Tethyan Orogenic Belt*. https://doi.org/10.5382/SP.19.13

Maepa, F., Smith, R. S., & Tessema, A. (2021). Support vector machine and artificial neural network modelling of orogenic gold prospectivity mapping in the Swayze greenstone belt, Ontario, Canada. *Ore Geology Reviews*, *130*, 103968. https://doi.org/10.1016/j.oregeorev.2020.103968

McInnes, L., Healy, J., & Melville, J. (2020). UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. arXiv:1802.03426

Melnik, O. E., & Bindeman, I. N. (2018). Modeling of trace elemental zoning patterns in accessory minerals with emphasis on the origin of micrometer-scale oscillatory zoning in zircon. *American Mineralogist*, *103*(3), 355–368. https://doi.org/10.2138/am-2018-6182

Miotto, R., Wang, F., Wang, S., Jiang, X., & Dudley, J. T. (2018). Deep learning for healthcare: Review, opportunities and challenges. *Briefings in Bioinformatics*, *19*(6), 1236–1246. https://doi.org/10.1093/bib/bbx044

Moral Cardona, J. P., Gutiérrez Mas, J. M., Sánchez Bellón, A., Domínguez-Bella, S., & Martínez López, J. (2005). Surface textures of heavy-mineral grains: A new contribution to provenance studies. *Sedimentary Geology*, *174*(3), 223–235. https://doi.org/10.1016/j.sedgeo.2004.12.006

Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted Boltzmann machines. In *Proceedings of the 27th International Conference on International Conference on Machine Learning* (pp. 807–814). Madison, WI: Omnipress.

Nasir, V., & Sassani, F. (2021). A review on deep learning in machining and tool monitoring: Methods, opportunities, and challenges. *International Journal of Advanced Manufacturing Technology*, *115*(9), 2683–2709. https://doi.org/10.1007/s00170-021-07325-7

Nathwani, C. L., Simmons, A. T., Large, S. J. E., Wilkinson, J. J., Buret, Y., & Ihlenfeld, C. (2021). From long-lived batholith construction to giant porphyry copper deposit formation: Petrological and zircon chemical evolution of the Quellaveco district, Southern Peru. *Contributions to Mineralogy and Petrology*, *176*(2), 12. https://doi.org/10.1007/s00410-020-01766-1

Ng, A. Y. (2004). Feature selection, L1 vs. L2 regularization, and rotational invariance. In *Proceedings of the Twenty-First International Conference on Machine Learning* (p. 78). New York, NY: Association for Computing Machinery. https://doi.org/10.1145/1015330.1015435

Palafox, L. F., Hamilton, C. W., Scheidt, S. P., & Alvarez, A. M. (2017). Automated detection of geological landforms on Mars using Convolutional Neural Networks. *Computers & Geosciences*, *101*, 48–56. https://doi.org/10.1016/j.cageo.2016.12.015

Perez, L., & Wang, J. (2017). The effectiveness of data augmentation in image classification using deep learning. arXiv:1712.04621.

Pires de Lima, R., Bonar, A., Coronado, D. D., Marfurt, K., & Nicholson, C. (2019). Deep convolutional neural networks as a geological image classification tool. *The Sedimentary Record*, *17*(2), 4–9. https://doi.org/10.2110/sedred.2019.2.4

Pires de Lima, R., & Marfurt, K. (2020). Convolutional neural network for remote-sensing scene classification: Transfer learning analysis. *Remote Sensing*, *12*(1), 86. https://doi.org/10.3390/rs12010086

Pizarro, H., Campos, E., Bouzari, F., Rousse, S., Bissig, T., Gregoire, M., & Riquelme, R. (2020). Porphyry indicator zircons (PIZs): Application to exploration of porphyry copper deposits. *Ore Geology Reviews*, *126*, 103771. https://doi.org/10.1016/j.oregeorev.2020.103771

Ranzato, M., Huang, F. J., Boureau, Y.-L., & LeCun, Y. (2007). Unsupervised learning of invariant feature hierarchies with applications to object recognition. In *2007 IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1–8). https://doi.org/10.1109/CVPR.2007.383157

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

Sainburg, T., McInnes, L., & Gentner, T. Q. (2021). Parametric UMAP embeddings for representation and semisupervised learning. *Neural Computation*, *33*(11), 2881–2907. https://doi.org/10.1162/neco_a_01434

Scherer, D., Müller, A., & Behnke, S. (2010). Evaluation of pooling operations in convolutional architectures for object recognition. In K. Diamantaras, W. Duch, & L. S. Iliadis (Eds.), *Artificial neural networks—ICANN 2010* (pp. 92–101). Berlin, Heidelberg: Springer.

Sheldrake, T., & Higgins, O. (2021). Classification, segmentation and correlation of zoned minerals. *Computers & Geosciences*, *156*, 104876. https://doi.org/10.1016/j.cageo.2021.104876

Shoji, D., Noguchi, R., Otsuki, S., & Hino, H. (2018). Classification of volcanic ash particles using a convolutional neural network and probability. *Scientific Reports*, *8*(1), 8111. https://doi.org/10.1038/s41598-018-26200-2

Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of Big Data*, *6*(1), 60. https://doi.org/10.1186/s40537-019-0197-0

Simmons, A. T. (2013). *Magmatic and hydrothermal stratigraphy of Paleocene and Eocene porphyry Cu-Mo deposits in southern Peru (PhD thesis)*. University of British Columbia.

Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, *15*(56), 1929–1958. http://jmlr.org/papers/v15/srivastava14a.html

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., et al. (2014). Going deeper with convolutions. arXiv:1409.4842.

Wager, S., Wang, S., & Liang, P. (2013). Dropout training as adaptive regularization. arXiv:1307.1493.

Waldeland, A. U., Jensen, A. C., Gelius, L.-J., & Solberg, A. H. S. (2018). Convolutional neural networks for automated seismic interpretation. *The Leading Edge*, *37*(7), 529–537. https://doi.org/10.1190/tle37070529.1

Wu, B., Li, X., Yuan, F., Li, H., & Zhang, M. (2022). Transfer learning and Siamese neural network based identification of geochemical anomalies for mineral exploration: A case study from the CuAu deposit in the NW Junggar area of northern Xinjiang Province, China. *Journal of Geochemical Exploration*, *232*, 106904. https://doi.org/10.1016/j.gexplo.2021.106904

Zou, S., Chen, X., Brzozowski, M. J., Leng, C.-B., & Xu, D. (2022). Application of machine learning to characterizing magma fertility in porphyry Cu deposits. *Journal of Geophysical Research: Solid Earth*, *127*, e2022JB024584. https://doi.org/10.1029/2022JB024584