DeepRock: Igneous Rock Image Classification

Jabs (Mohammad) Aljubran
Stanford University
aljubrmj@stanford.edu

Camilo Saavedra
Stanford University
csaavedr@stanford.edu

Abstract

State-of-the-art rock lithology identification is based on manual inspection and laboratory microscopic testing. This approach is time-consuming, expensive, uncertain, and subject to human bias. Furthermore, people always wonder about rock types during outdoor activities, so a rock classification tool would find many industrial and educational applications. Prior attempts to automate this process used various techniques without success. Yet, deep learning has not been considered, and thus this novel project explores this application. There is not an open-source dataset curated for this problem. Hence, we scraped the web and designed a dataset from scratch for igneous rock classification with 7 classes: Andesite, Basalt, Diorite, Gabbro, Granite, Peridotite, and Rhyolite. State-of-the-art pretrained computer vision architectures are explored, including VGG, ResNet, and DenseNet. Fully unfrozen, pretrained DenseNet121 showed the best performance on the validation set with average precision, recall, and F_1 score of 91.07%, 92.73%, and 91.17%, respectively. Error analysis and rock features are explored by generating Grad-CAM heatmaps to understand how the model arrives at class predictions. It is observed that granularity and color are major contributors to the classification process.

1. Introduction

Rock lithology identification is a vital component of field geological surveys. Traditionally, rocks are collected at the field and brought to laboratories to be identified and analyzed by experienced geologists. This process is expensive, time-consuming, uncertain, and subject to bias [16]. With the advent of deep learning in computer vision tasks, rock type identification, which represents the first step of the full geological survey process, can be improved and fully automated. Furthermore, autonomous rock classification will facilitate more discoveries in Earth geological investigations and Mars planetary surface exploration missions.

Indirect quantitative image analysis methods are common in analyzing rock samples, where the rock image is first segmented to allow for measuring various rock properties (size, granularity, mineralogy, orientation, etc.) [1][10]. With the aid of rock property books and glossaries, these features are used to identify the rock type. Other efforts involved the application of machine learning (support vector machine, K-nearest neighbors and decision trees) to classify the rock texture without knowledge of the type [12]. However, rock type identification is critical as it informs us about the geological history of the environment. For example, crystal size in igneous rocks reflects the cooling speed during rock formation time, while grain size and shape of sedimentary rocks indicate the type of the rock depositional environment [13].

2. Related Work

Many efforts focused on manual feature selection to encode the rock image before applying a classifier to identify the rock type. Efforts included the use of hand samples as well as microscopic samples with handcrafted spectral and textural features with different machine learning algorithms, e.g. K-nearest neighbors, optimal spherical neighborhood, Bayesian analysis, and linear discriminant analysis [7][8][15]. However, manual feature selection is not only time-consuming and hard to generalize, but also fails to accommodate for rock heterogeneity.

Hence, recent research efforts resorted to automatic feature generation to classify rock images. Using a dataset of 700 rocks with 9 types, researchers utilized semi-supervised feature detection based on K-means (~96% accuracy) and unsupervised self-taught feature encoding (~90% accuracy) to allow for rock classification with minimal manual effort in rock type labelling and feature extraction [13]. Other efforts involved using microscopic rock images with per-pixel classification of 4 intrusive igneous rock types based on edge and color features with over 90% accuracy [6]. Though these approaches would eliminate the need for an expert geologist if successfully generalized to more classes, they require microscopic images which is costly and time-consuming. Other efforts used a 6-layer convolutional neural network (CNN) architecture to classify rock granularity using microscopic rock imagery, yet this is still not a fully integrated approach and requires laboratory microscopic images [2].

This project aims to classify igneous rocks using hand sample images, which will make it usable for field research as well as educational and entertainment purposes by the

general public. Up to the authors' knowledge, this novel project is the first to explore the application of end-to-end CNN architectures to classify raw hand sample igneous rock images without the need for any expensive, time-consuming laboratory efforts.

3. Dataset

To the best of the authors' knowledge, there is no sufficiently large, labelled, and open-source rock image database. Hence, efficient data collection and preprocessing are critical components to this project. We collaborated with the Branner Earth Sciences Library & Map Collections staff and set up a public Dropbox account which was sent to the geology community within Stanford University to crowdsource labelled rock images. This option did not yield a sufficient dataset so far[1]. Hence, we scraped Google Images for rock images to build our own dataset[2].

Igneous rocks are generally classified based on two major criteria: grain size and mineral content. The most popular coarse-grained igneous rocks are granite, diorite, gabbro, and peridotite. Meanwhile, the most common fine-grained igneous rocks are rhyolite, andesite, and basalt. Hence, those will be the 7 classes of interest for this project, seen in **Fig. 1**.

Fig. 1—Popular Igneous Rock Types: Example images of the most common igneous rocks. Notice the variations in particle size and mineral content.

We scraped Google Images for 100 images per class, which were preprocessed in two stages. First, we manually eliminated the clearly mislabeled data, e.g. humans, trees, hills, etc. At this stage, the dataset suffered from class imbalance with image count of {Andesite [class 1]: 45, Basalt [class 2]: 42, Diorite [class 3]: 68, Gabbro [class 4]: 48, Granite [class 5]: 62, Peridotite [class 6]: 49, Rhyolite [class 7]: 52}. Secondly, we used an error analysis approach (to be highlighted in the upcoming sections) for further preprocessing, which was followed by scraping for more images to retain class balance. Hence, the final dataset size was 511 images with 73 images per class, which was split into train:validation:test ratios of roughly 80:10:10. Note that the small and imbalanced dataset imposes a considerable challenge towards the success of this project.

Image preprocessing further involved normalization of pixels using ImageNet dataset mean and standard deviation of (0.485, 0.456, 0.406) and (0.229, 0.224, 0.225), respectively [3]. Images are down-sampled to $224\ x\ 224$ to accommodate for these architectures. To alleviate the dataset small size issue, random data augmentation (horizontal flipping, vertical flipping, and 45° rotation) is applied to training examples.

4. Methodology

The Pytorch implementation is used to develop this end-to-end deep learning model[3]. This process involved exploring several computer vision architectures, transfer learning, and class activation mapping (CAM) for error analysis.

Weighted multiclass cross-entropy is the loss function of choice as it accounts for the dataset class imbalance [9]. Let n_i be the number of training examples of the i^{th}class, then the corresponding class weight is computed as $w_i = \frac{\max_i n_i}{n_i}$. Loss is minimized using stochastic gradient descent (SGD) with 0.9 momentum where it is found to generalize better for this task than adaptive techniques (AdaGrad, RMSProp, Adam, etc.) [17]. While loss and accuracy plots will be used to visualize the learning curves and fine tune the models, F_1 score is used as the evaluation metric to save a checkpoint of the best performing model during training. Average expressions of precision and recall are used to calculate average F_1 score since this is a multiclass task.

4.1. Architectures

After exploring simple CNN designs and experimenting with state-of-the-art computer vision models, three major architectures are nominated: VGG [14], ResNet [4], and DenseNet [5]. When compared to VGG, Resnet and Densenet incorporate residual and dense blocks, respectively, which alleviate gradient vanishing and allow for deeper network designs by facilitating gradient flow during backpropagation. Different depths and layer count of these architectures are explored as part of the optimization process.

Note that the hyperparameter tuning process is not explicitly outlined due to space limitation. However, it is important to note that learning rate step decay (factor of 0.5 every 10 epochs) is used to train these models for a total of

[1] We are still open to accept more labelled rock images at: https://www.dropbox.com/request/6oNyy0Eb4pexSC0py8Jt

[2] Github repository that we used to scrape Google Images: https://github.com/hardikvasa/google-images-download.git

[3] Github repository of our work and implementation can be found at: https://github.com/aljubrmj/CS230-Deep-Learning-Project

50 epochs using batch size of $\{4, 8, 16, 32, 128\}$. In addition, regularization (data augmentation, dropout, L2 weight decay, and early stopping) is applied to prevent overfitting the training dataset and handle the bias-variance tradeoff. dropout probability and L2 regularization strength spanned across $\{0.2, 0.5, 0.8\}$, and $\{0.0001, 0.001, 0.01\}$, respectively. Linear and log grid searches are used for different hyperparameters as appropriate.

Transfer learning framework is incorporated to allow for leveraging knowledge (features, weights, etc.) of other image recognition tasks, which improves model performance and training speed. ImageNet-pretrained model weights are used as initialization in various forms. Note that rocks are characterized by granularity which is microscopic, unlike the macroscopic nature of the ImageNet features. Hence, unfreezing the last affine layer is insufficient to achieve the optimal results. Rather, training must involve unfreezing convolutional layers even though the training set at hand is small.

Architecture description is limited to DenseNet121 as the results will show that it is the top-performing model. This architecture connects each layer to every other layer in a feed-forward fashion to form dense blocks, seen in **Fig. 2**. Each layer receives the preceding feature maps as input while passing its own feature maps to all subsequent layers.

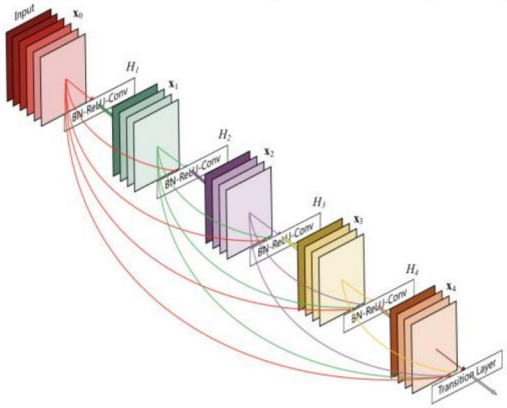

Fig. 2—DenseNet Dense Block: Illustration of a 5-layer dense block with growth rate of 4 [5].

Transition convolutional and max pooling layers are used to connect these dense blocks and manipulate feature map sizes. Each convolutional layer is designed to perform batch normalization, rectified linear unit, and convolution (BN-ReLU-Conv) operations in order, denoted by $H(.)$. For a DenseNet121 architecture with L layers, there are $\frac{L(L+1)}{2}$ direct connections compared to L direct connections traditionally. Let the output of the l^{th} layer be x_l, **Eq. 1** shows how layers are connected in dense blocks. This architecture is advantageous as it alleviates the vanishing-gradient problem, accelerate optimization, and significantly reduce the number of parameters.

$$x_l = H_l([x_0, x_1, \dots, x_{l-1}]) \qquad (\boldsymbol{Eq.1})$$

4.2. Error Analysis

Error analysis is important to understand what features the model looks for to make predictions, find out which class results in most mispredictions, and filter out the training dataset. Gradient-weighted Class Activation Mapping (Grad-CAM) is used for this purpose.

Grad-CAM is used to generate a coarse localization heatmap highlighting the important regions or pixels in the image which the model used to generate its prediction [11]. While the original CAM paper [18] requires the substitution of fully-connected layers with a global average pooling layer after the convolutional blocks, Grad-CAM is advantageous as it is applicable to a significantly broader range of architectures, as seen in **Fig. 3**. This technique will contribute significantly to further preprocessing and curating the dataset.

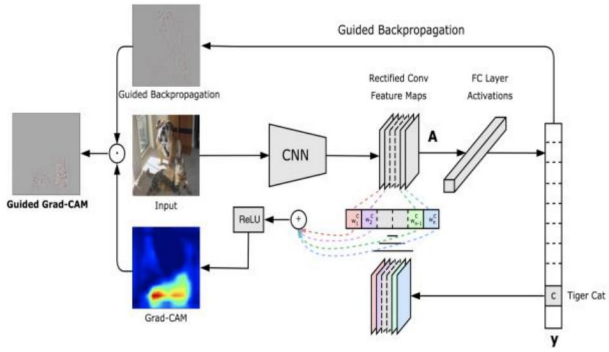

Fig. 3—Grad-CAM Workflow: Image is forward-propagated through the CNN to obtain raw scores. The gradient of the desired class is set to 1 while all others are strictly set to zero. The signal is then back-propagated to rectified convolutional feature maps which generate the coarse localization [11].

5. Preliminary Results

With 7 classes at hand, random guessing results in cross-entropy loss and accuracy of 1.946 and 14.29%, respectively. As seen in **Fig. 3**, randomly initialized ResNet18 architecture is used to first build a baseline model and overfit the training data that outperforms random guessing. Moreover, using a fully frozen pretrained ResNet18 while only training on the last affine layer yields further improvement. These two models resulted best validation accuracy of 51.78% and 66.07%, respectively.

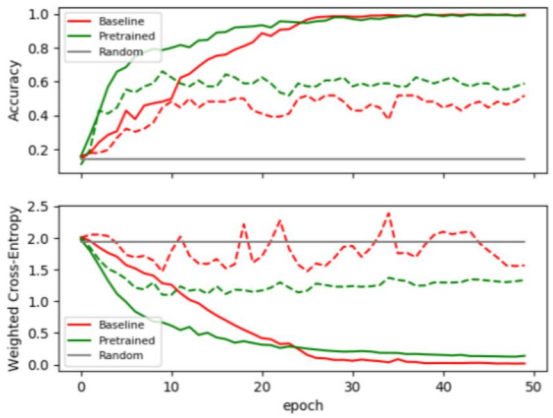

Fig. 3—Preliminary Models: Comparison of training (solid line) and validation (dashed line) loss and accuracy of randomly initialized and pretrained ResNet18 models. Note that using fully frozen pretrained models while only training the final affine layer results in significant improvement.

Given the added value of transfer learning, the three aforementioned state-of-the-art computer vision architectures are evaluated. All are initialized with ImageNet-pretrained weights with fully frozen layers, except for the last affine layer. Meanwhile, hyperparameters are tuned and regularization is applied. As seen in **Fig. 4,** VGG19 with batch normalization (VGG19_BN), ResNet18, and DenseNet121 are found to be the best-performing with best validation accuracies of 66.07%, 71.43%, and 78.57%, respectively. Note that DenseNet121 resulted in the top performance while both DenseNet121 and ResNet18 trained faster than VGG19_BN since the latter has significantly higher number of weights feeding into the last affine layer.

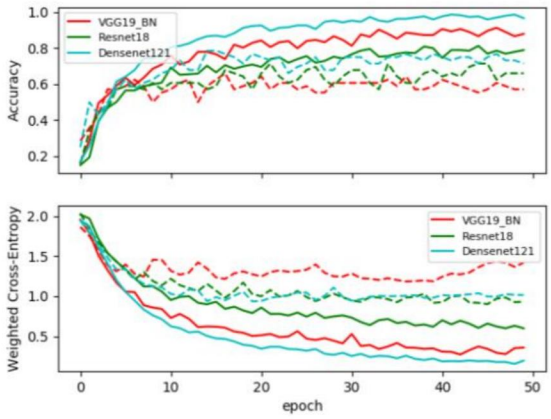

Fig. 4—Pretrained Models: Comparison of training (solid line) and validation (dashed line) loss and accuracy of pretrained models. Note that these models are fully frozen, except for the final affine layer. DenseNet121 outperforms all others in validation accuracy.

To further improve the performance, we inspected the class precision and recall for the highest performing model, seen in **Table 1**. Note that the model mispredictions are mainly due to andesite, diorite, and gabbro (classes 1, 3, and 4, respectively). Observing these rocks in the dataset, see Fig. 1, they actually do look quite similar in granularity and color. A human would definitely struggle to differentiate between these rocks. Meanwhile, observing the imbalanced nature of the dataset, there is no relationship between class F_1 score and class sample size. This is attributed to the use of the weighted cross-entropy loss which algorithmically alleviates the class imbalance problem.

Table 1—Model Metrics: precision, recall, and F_1 score

Class	1	2	3	4	5	6	7
Precision	0.75	0.75	0.75	0.63	0.75	1.00	0.88
Recall	0.67	0.86	0.67	0.71	1.00	0.73	1.00
F1 Score	0.71	0.80	0.71	0.67	0.86	0.84	0.93

To further analyze the model error, it is important to inspect the features (pixels) which trigger the model to predict the output rock class. Grad-CAM is used to generate heatmaps that indicate the discriminative image regions used by the model to identify a specific category. As seen in **Fig. 5**, significant insights regarding the dataset quality are drawn which explain most of the model mispredictions. These images are drawn from the training dataset, so they reflect what the model is learning to classify each specific rock type. We can see that the image object count, background color, and illumination level significantly affect the model classification decision.

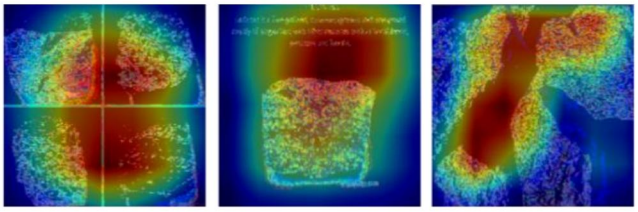

Fig. 5—Erroneous Dataset Images: Grad-CAM heatmaps of erroneous images drawn from the training dataset, showing how the model is influenced by image illumination level, background color, and object count.

Hence, Grad-CAM heatmaps are used to further curate the dataset, which typically exacerbated the data imbalance problem. As a result, we scraped the web for more data but this time with a priori knowledge of the malignant image variations that need to be avoided. This represents the last preprocessing step, and it brings up the final dataset size to 511 images with 73 images per class. Therefore, the dataset is fully balanced with properly curated images.

VGG19_BN, ResNet18, and DenseNet121 were all retrained after this dataset curation step. In addition to hyperparameter tuning and only unfreezing the final affine layer while transfer learning the features from ImageNet, we gave these architectures a chance to train without freezing any layers. In other cases, we froze all layers except for the all affine layers as well as a convolutional

layer. This choice is based on the intuition that shallower layers learn simple features, e.g. edges, while deeper layers learn more complex shapes. The nature of rock features (granularity, crystal shape and size, heterogeneity, etc.) are microscopic which is not fully captured from the mainly macroscopic features extracted from ImageNet. Despite this difference in the two tasks, using ImageNet-pretrained models is still advantageous, as shown in Fig. 3.

Fig. 6 shows model retraining results with significantly higher best validation accuracy. Using the ImageNet-pretrained weights for initialization only while leaving all architecture layers unfrozen, DenseNet121 is the top-performing model with best validation accuracy of 91.07%. Note that this model checkpoint is stored based on the highest F_1 score recorded during training. This result emphasizes the fact that the ImageNet features do not fully capture the rock features, and some or all of the convolutional layers need to be unfrozen to capture these missing features despite the small dataset size.

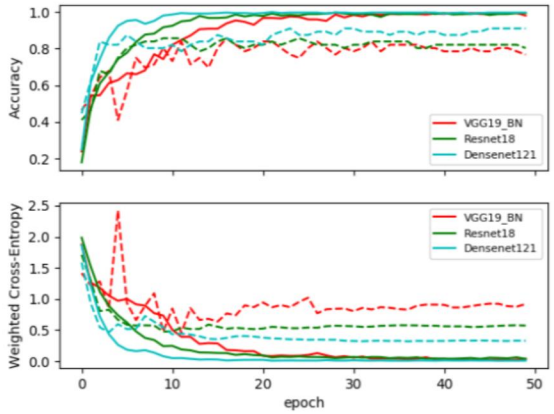

Fig. 6—Unfrozen Models: Comparison of training (solid line) and validation (dashed line) loss and accuracy of pretrained models. Note that these models incorporate unfrozen convolutional layers besides unfreezing affine layers, which allows for learning rock-specific features. DenseNet121 outperforms all others in validation accuracy.

Model performance on the validation set is further evaluated using confusion matrix, precision, recall, and F_1 score, seen in **Table 2**. Note that the validation set contains exactly 8 images of each class. The average precision, recall, and F_1 score are 91.07%, 92.73%, and 91.17%, respectively. Despite curating the dataset, the model still incurs most mispredictions due to andesite, diorite, and gabbro (classes 1, 3, and 4, respectively). Again, this result is anticipated as these rocks are visually challenging to differentiate. Further increasing the size of the dataset with well curated example is expected to alleviate this problem. Meanwhile, evaluating the model on the test set which also contains 8 rocks per class resulted in accuracy, precision, recall, and F_1 score of 75.00%, 75.00%, 81.63%, and 78.17%, respectively.

Table 2—Final Model Validation Evaluation Metrics: confusion matrix, precision, recall, and F_1 score

Class	1	2	3	4	5	6	7
1	6	1	0	1	0	0	0
2	0	7	0	1	0	0	0
3	0	0	7	0	1	0	0
4	0	0	0	8	0	0	0
5	0	0	0	0	8	0	0
6	0	0	0	1	0	7	0
7	0	0	0	0	0	0	8
Precision	0.75	0.88	0.88	1.00	1.00	0.88	1.00
Recall	1.00	0.88	1.00	0.73	0.89	1.00	1.00
F1 Score	0.86	0.88	0.93	0.84	0.94	0.93	1.00

It is important to remember that these rock glossaries and categories are invented by humans after all, and do not necessarily reflect the optimal rock categorization. Hence, rock glossaries could also be reviewed based on the interpretation insights acquired from deep learning models. **Fig. 7** shows Grad-CAM maps generated using the highest performing DensNet121 model after fully curating the dataset. Although the dataset is small, we can evidently see that the network is actually capturing granularity features which is a critical component to physical rock properties.

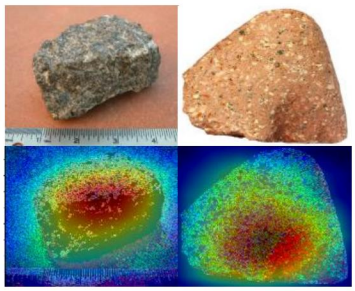

Fig. 7—Final Grad-CAM Heatmap: comparing the Grad-CAM heatmaps of the coarse-grained Gabbro (left) and fine-grained Rhyolite (right), we observe that the model focuses on granular features to make its predictions.

In addition, we attempted greyscale data augmentation to understand the reliance of model predictions on color. We found that validation performance dramatically dropped below 15% which is almost at the random guessing level. Hence, color is a crucial feature to the network even though some rock classes can take different colors as seen in the dataset.

6. Future Work

Several techniques and approaches can be explored to further improve this classification model:

- Incorporate microscopic images to the dataset for feature extraction purposes, yet they should not be used to build the final model since the goal is to predict rock type using hand samples
- Explore transfer learning of features learned from datasets with more microscopic features other than ImageNet

- Consider a pipeline model with an attention layer module to avoid confusing the rock due to variations in background, illumination, etc.
- Expand this work to more igneous rock types as well as sedimentary and metamorphic rocks.

7. Contributions

Jabs and Camilo contributed equally to all tasks involved in completing this project, including data collection and preprocessing, code development, model training and validation, evaluation, result analysis, and report writeup.

References

[1] Chanou, A., Osinski, G. R., & Grieve, R. A. F. (2014). A methodology for the semi-automatic digital image analysis of fragmental impactites. Meteoritics & Planetary Science, 49(4), 621-635.

[2] Cheng, G., & Guo, W. (2017, August). Rock images classification by using deep convolution neural network. In Journal of Physics: Conference Series (Vol. 887, No. 1, p. 012089). IOP Publishing.

[3] Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., & Fei-Fei, L. (2009, June). Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition (pp. 248-255). Ieee.

[4] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).

[5] Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4700-4708).

[6] Joseph, S., Ujir, H., & Hipiny, I. (2017, September). Unsupervised classification of Intrusive igneous rock thin section images using edge detection and colour analysis. In 2017 IEEE International Conference on Signal and Image Processing Applications (ICSIPA) (pp. 530-534). IEEE.

[7] Lepistö, L., Kunttu, I., Autio, J., & Visa, A. (2003). Rock image classification using non-homogenous textures and spectral imaging.

[8] Młynarczuk, M., Górszczyk, A., & Ślipek, B. (2013). The application of pattern recognition in the automatic classification of microscopic rock images. Computers & Geosciences, 60, 126-133.

[9] Panchapagesan, S., Sun, M., Khare, A., Matsoukas, S., Mandal, A., Hoffmeister, B., & Vitaladevuni, S. (2016, September). Multi-Task Learning and Weighted Cross-Entropy for DNN-Based Keyword Spotting. In *Interspeech* (pp. 760-764).

[10] Pittarello, L., & Koeberl, C. (2013). Clast size distribution and quantitative petrography of shocked and unshocked rocks from the El'gygytgyn impact structure. Meteoritics & Planetary Science, 48(7), 1325-1338.

[11] Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2017). Grad-cam: Visual explanations from deep networks via gradient-based localization. In Proceedings of the IEEE International Conference on Computer Vision (pp. 618-626).

[12] Shang, C., & Barnes, D. (2012, June). Support vector machine-based classification of rock texture images aided by efficient feature selection. In The 2012 International Joint Conference on Neural Networks (IJCNN) (pp. 1-8). IEEE.

[13] Shu, L., McIsaac, K., Osinski, G. R., & Francis, R. (2017). Unsupervised feature learning for autonomous rock image classification. Computers & Geosciences, 106, 10-17.

[14] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

[15] Singh, M., Javadi, A., & Singh, S. (2004, August). A comparison of texture features for the classification of rock images. In International Conference on Intelligent Data Engineering and Automated Learning (pp. 179-184). Springer, Berlin, Heidelberg.

[16] Wang, C., Li, Y., Fan, G., Chen, F., & Wang, W. (2018). Quick Recognition of Rock Images for Mobile Applications. Journal of Engineering Science & Technology Review, 11(4).

[17] Wilson, A. C., Roelofs, R., Stern, M., Srebro, N., & Recht, B. (2017). The marginal value of adaptive gradient methods in machine learning. In *Advances in Neural Information Processing Systems* (pp. 4148-4158).

[18] Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., & Torralba, A. (2016). Learning deep features for discriminative localization. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2921-2929).