Towards Building a Geological Cyber-infrastructure: Classifying Sigma-clast Images in Photomicrographs

Boaz Cogan, Mathew Puryear, Dr. Gurman Gill
Department of Computer Science, Sonoma State University, Rohnert Park

Introduction

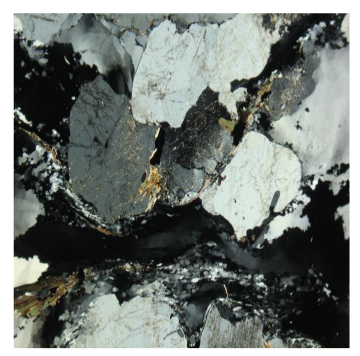
Analog means of collecting, storing, and analyzing geological data are outdated and it has become necessary to create a cyberinfrastructure to record observations and compare findings. The overarching goal is to create a system that users can provide an image and, based on automated analysis of this image, get immediate feedback such as images of similar photomicrographs. The purpose of this study is to find which, if any, machine learning algorithm can accomplish this. Both "traditional" machine learning and recent "deep learning" algorithms have been evaluated and found to be promising for classifying photomicrographs containing sigma-clast structures.

Dataset

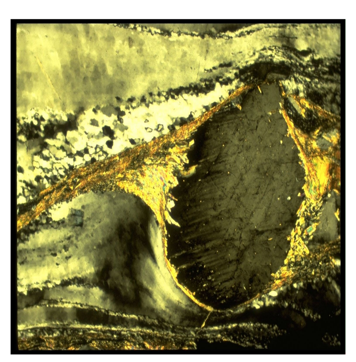
In this experiment there are 100 images of sigma-clasts as well as 100 non-sigma clast images, provided by the SSU geology department. Additional images of geological formations have been parsed from online sources to bolster the non-sigma-clast dataset. Since this is a binary classification experiment there are only two classes: sigma-clast and non-sigma-clast.


Sigma-Clast
Confidence Score: 0.757


Sigma-Clast
Confidence Score: 0.711


Non-Sigma-Clast
Confidence Score: 1.00


Sigma-Clast
Confidence Score: 1.00


Sigma-Clast
Confidence Score: 0.718


Non-Sigma-Clast
Confidence Score: 0.999

Training Convolutional Neural Networks (CNN) requires vastly more images than is available in the current data set. To increase the size of the dataset data augmentation is employed, as explained next.

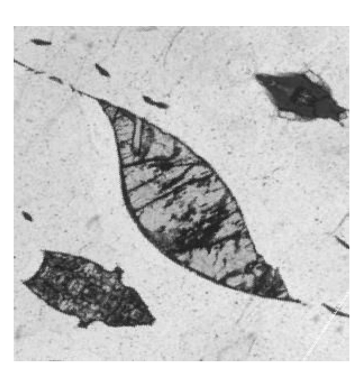
Data Augmentation

Since there is not a large dataset of sigma-clasts readily available to utilize in training a neural network. We utilized data augmentation in order to bolster our dataset to something that would be usable. Data augmentation is the process of taking the images one currently has, and manipulating them such that they are a different image while still maintaining their label. The techniques utilized to accomplish this are: rotation, shear, shift, zoom, and reflection.

Effects Utilized:
rotation range: 40 degrees
height/width shift: 10%
Shear range: 0.2 radians ccw
Zoom: 30%
Horizontal flipping: true
Fill mode : "reflect"


Counterclockwise rotation
zoom


Height Shift
Width Shift
Zoom

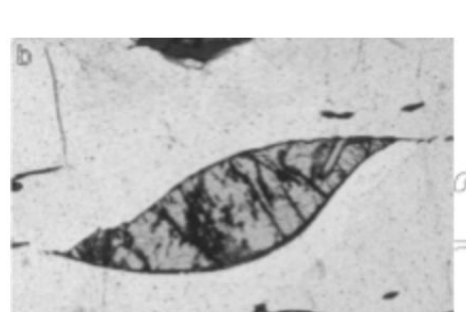

Horizontal Flip
Clockwise Rotation
Filled

Traditional Machine Learning

To establish a frame of reference for the applicability of convolutional neural networks to the provided dataset, one must first evaluate the efficacy of traditional machine learning techniques. Our team applied the image classification problem to the Bag of Words model in collaboration with a Support Vector Machine (SVM).

Support Vector Machines

SVM's attempt to separate data into two distinct subsets using a hyper-planar linear classifier using a set of provided features. We extracted SiFT descriptors from a given image and used the features as input to an SVM. We primarily utilized the SVM model to evaluate the applicability of most of our given models.


Input Sigma-Clast Image

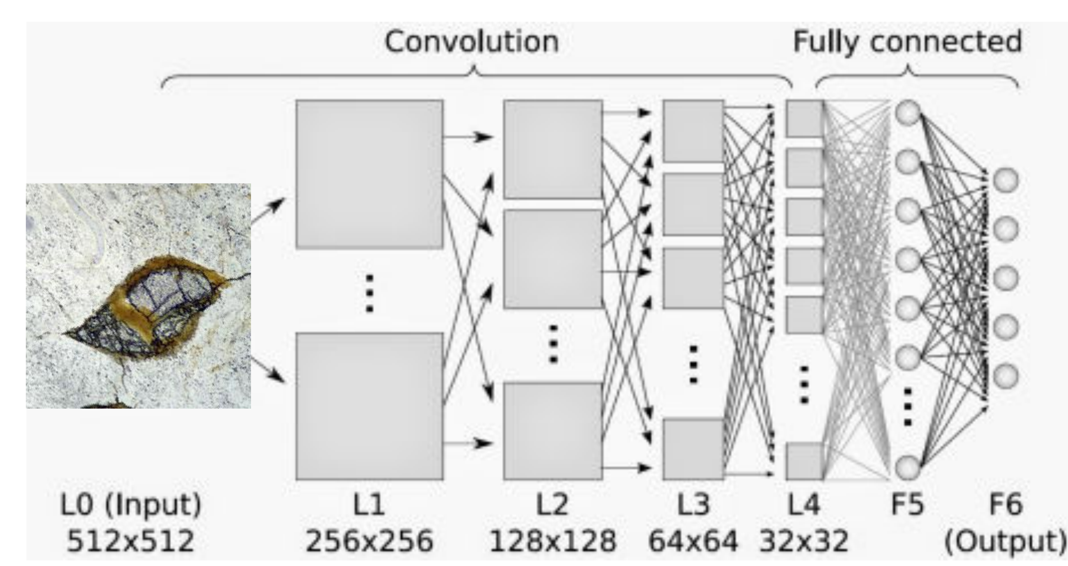

After Canny Edge Detection


After SiFT Feature Detection

Convolutional Neural Networks (CNN)

Introduction

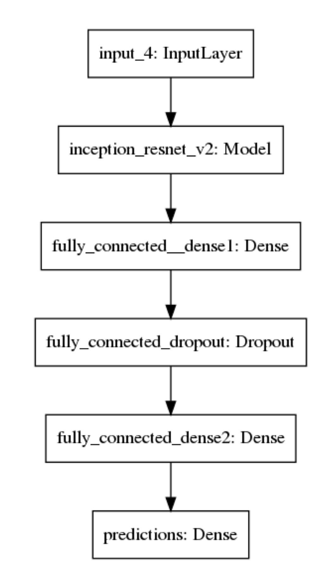
A Convolutional Neural Network (CNN) takes as input an image, and applies collections of filters to the image. As one progresses through the provided figure, from left to right, filters in the layers(L1, L2, L3, L4) depict more complex structures. At layer L1, a CNN would detect generic shapes and edges, where as layer L4 may look for specific structures, such as the sigmoidal shape of a sigma-clast.



CNN Example

Transfer Learning

The experiment begins with testing the accuracy of transfer learning, utilizing popular convolutional neural network structures. Transfer learning is the process of taking a CNN that has been previously trained on a large, dataset and then applying that model to a different, usually smaller, dataset. To accomplish transfer learning we take a CNN, in our example we used InceptionResNetV2, remove the provided fully connected neural network, and we replace it with our own fully connected neural network. After the fully connected layer is replaced, we train and test using our own dataset. The fully connected model we chose had a 4096 node fully connected layer 1 followed by a 50% dropout layer to help reduce simply memorizing our small dataset. That layer is followed by a 1024 node fully connected layer. Lastly there is a prediction layer with a single node used for binary classification.

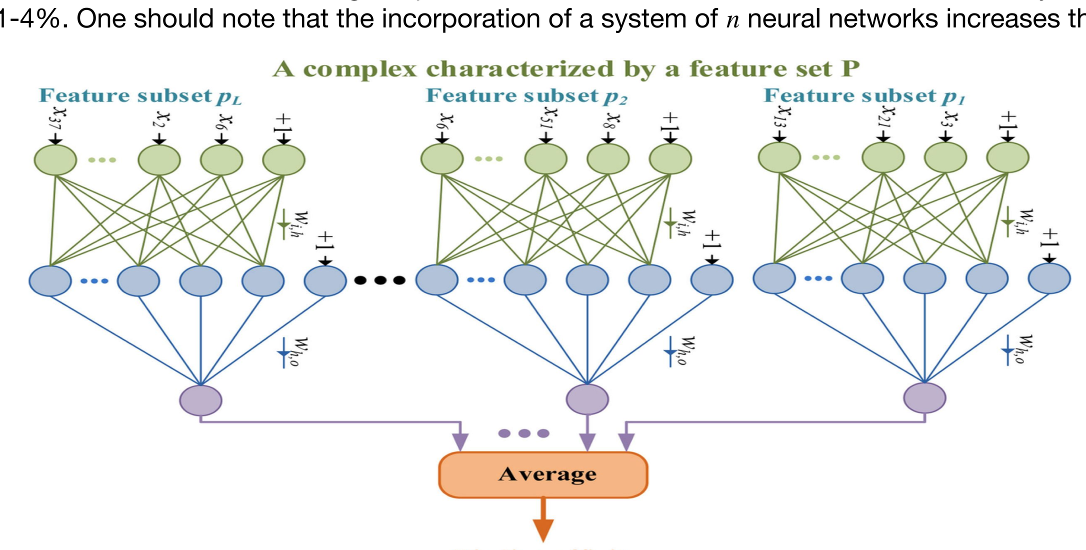


Fine Tuning

Fine tuning a model is the act of retraining specific layers in the pretrained CNN. The idea of fine tuning is that one is looking to retrain the filters used to find features in images. By manipulating the end layers of a CNN, and not the initial layers, one maintains the value gained of having a pretrained network and gain the ability to detect complex features more specific to an input dataset. Using the CNN Example above, one would enable training on some layers L4 through L1, but training on all 4 layers of the above CNN would be similar to training a CNN from scratch. We noticed diminishing returns on the number of layers we enabled training on, after more than just a few layers, ~4, our increase in accuracy was seemingly negligible.

Neural Network Ensembles

Two minds are better than one. Ensembles of neural Networks utilize this idea, assuming that different images are being consistently misclassified by each network, the mistake of one network may be caught and corrected by another. This is accomplished by simply averaging the final output of n trained networks before thresholding the prediction. We have found that the boost in accuracy is most significant when $n>2$ with an increase between 1-4%. One should note that the incorporation of a system of n neural networks increases the runtime of the algorithm by a factor of n.



Evaluation

When each model was tested for validity we used a stratified k-fold cross validation methodology, and measured the mean accuracy across all k folds tested. When testing the ensemble of networks as well as any individual network we utilized the split of data shown on the right. Two metrics were primarily used in the evaluation of our systems: the accuracy and F1 score of the network or system of networks. To calculate these metrics one needs the amount of images correctly classified in the positive or negative class, TP and TN, and the amount of images incorrectly classified in the positive class or negative class, FP and FN. The accuracy is defined as: (TP+TN)/(The total number of images). The F1 score is defined as: 2TP/(2TP+FP+FN).

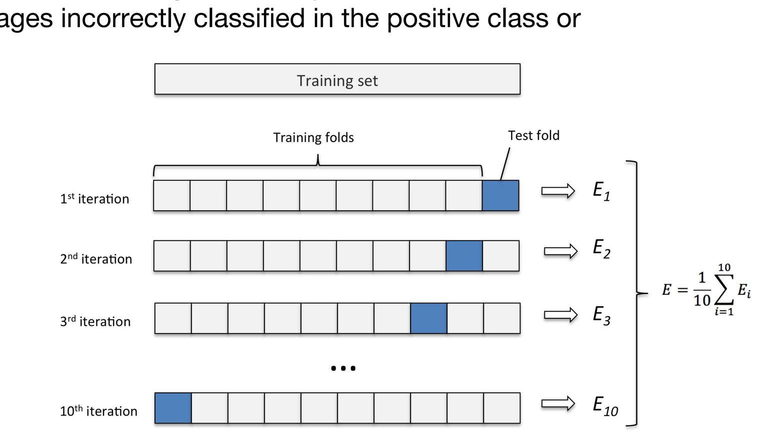
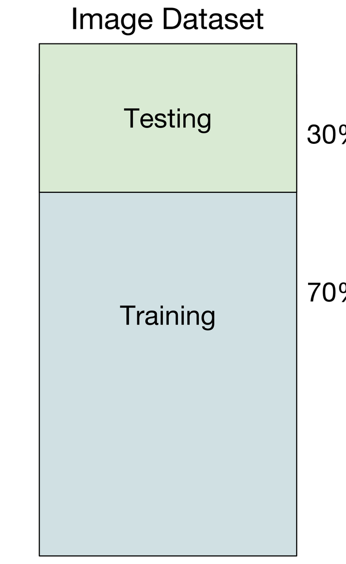
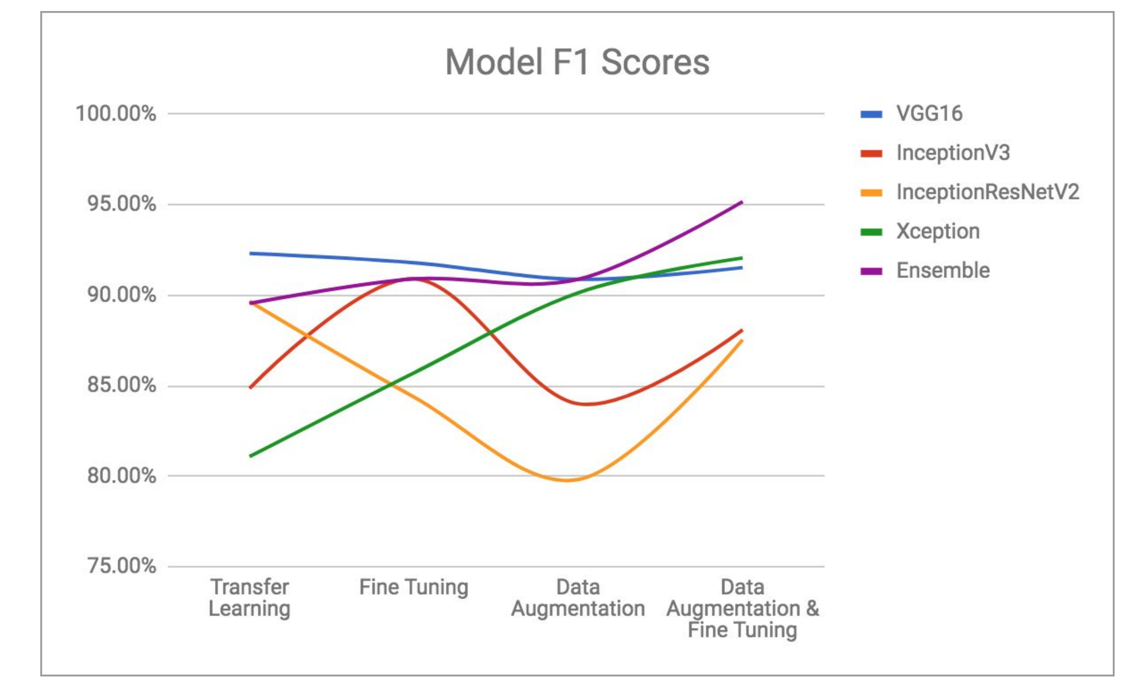


Image Dataset



Results

The best results were found when training four networks: VGG16, InceptionV3, InceptionResNetV2, and Xception. The accuracy of the individual networks while using both data augmentation and fine tuning were: 92.03%, 87.47%. 86.72%, and 92.4% respectively. The achieved accuracy of the ensemble of networks was 95.25%. This is notably better than the result of the traditional machine learning model which was able to recognize 85% of the sigma-clast images.

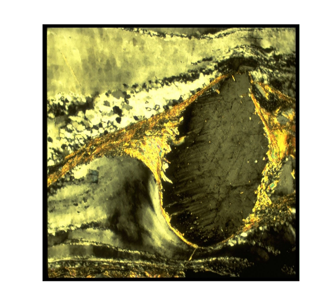


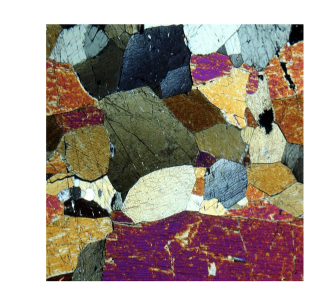
		Predicted							
		Transfer Learning		Fine Tuning		Data Augmentation		Data Augmentation & Fine Tuning	
		Non-Sigma	Sigma	Non-Sigma	Sigma	Non-Sigma	Sigma	Non-Sigma	Sigma
Actual	Non-Sigma	30	0	30	0	254	9	247	16
	Sigma	7	23	6	24	42	222	9	255

Ensemble Confusion Matrices

Result Analysis

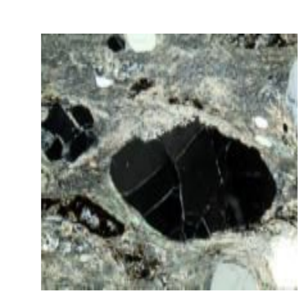
Below are some images that were commonly incorrectly classified by traditional machine learning methods and their ensemble classifications and confidence scores. The far left image, while a sigma-clast, has it's head protruding off the edge of the image, and is thus difficult to process. The center image contains no sigma-clast, but has many sigma-clast-esque shapes. The far right image is a sigma-clast which may have been incorrectly classified by traditional machine learning due to the granularity of the background.


Actual Class: Sigma
Traditional Predicted Class: Non-Sigma
Ensemble Predicted Class: Sigma
Ensemble Confidence Score = 0.7179

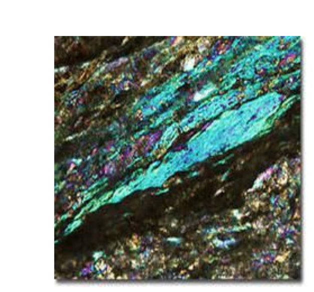

Actual Class: Non-Sigma
Traditional Predicted Class: Sigma
Ensemble Predicted Class: Non-Sigma
Ensemble Confidence Score = 0.999


Actual Class: Sigma
Traditional Predicted Class: Non-Sigma
Ensemble Predicted Class: Sigma
Ensemble Confidence Score = 1

Despite the increase in accuracy for the ensemble of networks, some images were still consistently being misclassified. The far left image is very similar to a sigma-clast, in that it has a rounded shape, but it is missing a head/tail. The center image is predicted to not contain a sigma-clast when it does contain one. This might be caused by data augmentation modifying an image to such an extent that it no longer contains a sigma-clast. Since the far right image was scraped from the web to bolster our non-sigma-clast data, it is possible that it does contain a sigma-clast and that the label is incorrect.


Actual Class: Non-Sigma
Ensemble Predicted Class: Sigma
Ensemble Confidence Score = 0.506


Actual Class: Sigma
Ensemble Predicted Class: Non-Sigma
Ensemble Confidence Score = 0.6


Actual Class: Non-Sigma
Ensemble Predicted Class: Sigma
Ensemble Confidence Score = 0.656

Conclusion and Future Research

The application of convolutional neural networks to the classification of sigma-clasts has provided promising results, performing significantly better then the best of the applied traditional machine learning models. Furthermore, most networks showed a upward trend in accuracy as more complex experiments were tested. Future work includes the implementation of a hybrid of techniques, utilizing the feature extracting capabilities of the convolutional neural network as input to an SVM as well as implementation of attention networks to provide better analysis of our results.

Acknowledgements:

We would like to thank the Koret foundation for funding our research and travel, as well as Dr. Matty Mookerjee for providing us with sigma-clast data.