Detection of Mineral Deposits in NASA *LandSat* Images Using Convolutional Neural Networks

Sean McAuliffe, V00913346 **Spencer Davis, V00759537** **Kiana Pazdernik, V00896924**

Mateo Moody, V00918050 **Chris Wong, V00780634**

Abstract

We trained Convolutional Neural Networks (CNNs) to identify locations likely to contain useful mineral deposits on the Earth's surface using NASA Landsat Earth observation images. Training labels for supervised learning were generated programmatically using a corresponding dataset of known mineral deposits. AI assistance in locating mineral-rich locations can potentially help to minimize the environmental impact caused by prospecting and mineral extraction, which is generally harmful to the environment and human health. The Landsat dataset is preprocessed by downsampling to a resolution of 512x512. The images are labelled using a K-D Tree Search Algorithm into the minerals dataset. Both binary classification and regression labels were generated for all images. Sets of labelled images are organized into buckets based on their location. Ocean masking is performed to remove buckets over open water to prevent the resulting ML agent from simply learning to distinguish between land and sea. The images at each location are then sorted by degree of cloud cover, allowing for the selection of training images with minimal cloud cover. Once the images are preprocessed, they are split into training and testing sets for supervised learning. Several combinations of model architectures and hyperparameters were investigated to find a model which achieved high accuracy. For evaluation purposes, all models were compared against a baseline obtained by training the model on a training set with randomized labels. The trained models obtained an accuracy of 73% on the binary classification problem, and a MAE of 0.117 on the regression problem; these values both represent notable improvements over baseline. Training on more powerful hardware and using the full-resolution images may produce further improvements.

1 Introduction

1.1 Problem Description

The work of prospecting for mineral deposits is complicated and can be dangerous. In some cases it can be harmful to the environment and human health. Current techniques depend on specific geological knowledge, and utilize a combination of magnetic, gravimetric radiometirc, and seismic methods. Our work proceeds from curiosity; can AI assistance be used to learn generalizable patterns — as of yet unknown to human geologists — that emerge in surface features, which may be highly predictive in identifying useful mineral deposits?

1.2 Approach

This project was conceived, in part, to take advantage of the extensive Landsat dataset, which is freely available and contains over 8 million Earth observation scenes taken over the past 50 years [1].

Project GitHub: https://github.com/SeanMcAuliffe/satimg-classifier

Landsat images are very high resolution, as such, Convolutional Neural Networks were chosen as the primary learning agent for this project.

Convolutional Neural Networks have proven to be very effective in image classification tasks [2]. Of particular interest to this project is the ability of CNNs to quickly reduce the resolution of an image, while preserving learned patterns. This is useful for our project, as the Landsat images are very high resolution, and would be difficult to train on using a consumer GPU.

The Landsat dataset was combined with a dataset of known mineral deposits to generate training labels for supervised learning; the label of each image indicated the presence or richness of minerals in the image.

Much of the work involved in this project was in the preprocessing and problem setup. These steps anticipated and attempted to remove sources of error, and to constrict the the problem such that the ML agent would learn to identify useful patterns. For example, algorithms were devised to identify and select training images having a minimal level of cloud cover, and to exclude images taken over water. The full details of the preprocessing pipeline can be found in section 2.

1.3 Goals

We aim to train Convolutional Neural Networks to achieve 80% validation accuracy on the binary classification problem, and a Mean Absolute Error of ≤ 0.1 on the regression problem.

2 Problem Forumulation

The problem formulation step encompases the bulk of the work. This step prepares the data for the model to train on. The particular ways in which the dataset is chosen, labelled, and prepared effect what the model will learn. These decisions are made with the goal of constraining the problem in such a way that the model will learn to solve the problem as formualted in our minds. Each formulation rests on a particular philosophy of what should be learned, and each has its own strengths and weaknesses.

We formulated the problem in two ways: binary classification, and regression. Each is described in the following subsections. Both approaches shared much of the same acquisition and preprosccesing pipeline as described below.

2.1 Data Acquisition & Preprocessing

We acquired the entire set of approximately 45000 band 7 images from Landsat 4 of the public Landsat dataset on Google Cloud Platform. We chose band 7 (2.09-2.35μm) because it uses a Thematic Mapper sensor to capture light in frequencies optimal for identifying hydrothermally-altered rocks associated with mineral deposits. These images are each approximately 7500x7500 pixels in a single colour channel, and are taken at regular locations on a 2x2° latitude-longitude grid.

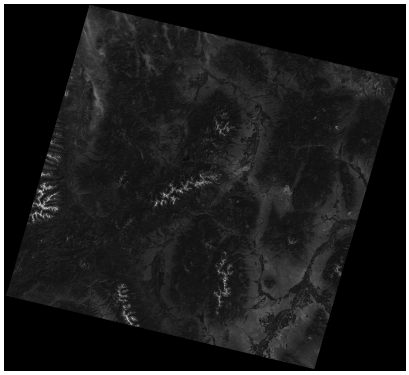
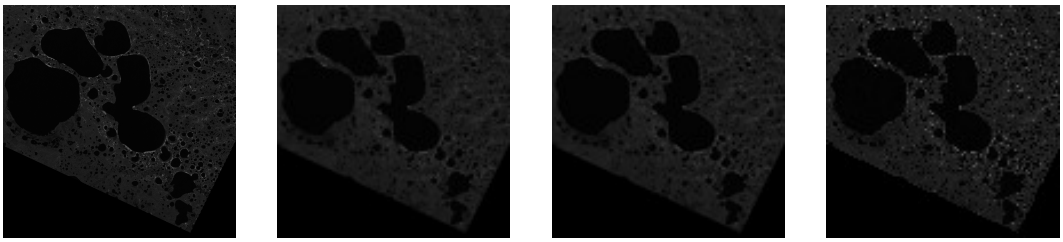


Figure 1: *An example Landsat image in band 7.*

We also acquired a dataset of approximately 350000 mineral deposits, each including the deposit's coordinates and the types of minerals present.

We automated the image downloads using a bash script. This script used the bash utility *ImageMagick* to downsample each image from 7500x7500 to approximately 512x512, to allow for training on consumer hardware. The corresponding metadata file for each downloaded image was also obtained, containing information such as the coordinates of the image corners. The download took approximately 6 seconds per image, for a total download time of approximately 75 hours.



(a) *Original* (b) *Rescaled* (c) *Resized* (d) *Downsampled*

Figure 2: Three compression techniques applied to the same Landsat image

Several differen algorithms from were experimented with: resizing, rescaling, and sampling. Resizing and rescaling use similar iterative processes of averaging neighboring pixels to effectively compress the image while reducing quality. Sampling is a more direct approach which simply selects a subset of the pixels evenly from across the input image. This is the lossiest of the three methods, but is also the fastest. The results of these three methods are shown in Figure 2.

Labels were created for each image using an algorithm accelerated by a k-d tree of mineral deposit locations; the k-d tree was used since k-d trees provide significant speedup for queries on spatial data [3]. Efficient label creation was nontrivial since it required identifying which mineral deposits were present in each image, and the images were neither axis-aligned nor perfectly rectangular. To address this, the label-creation algorithm first computed the enclosing circle of each image, then queried the k-d tree for the set of deposits present in the circle; brute force was then used to test which of these deposits were within the convex polygon of the image. This method provided a speedup of 20x over the naive approach.

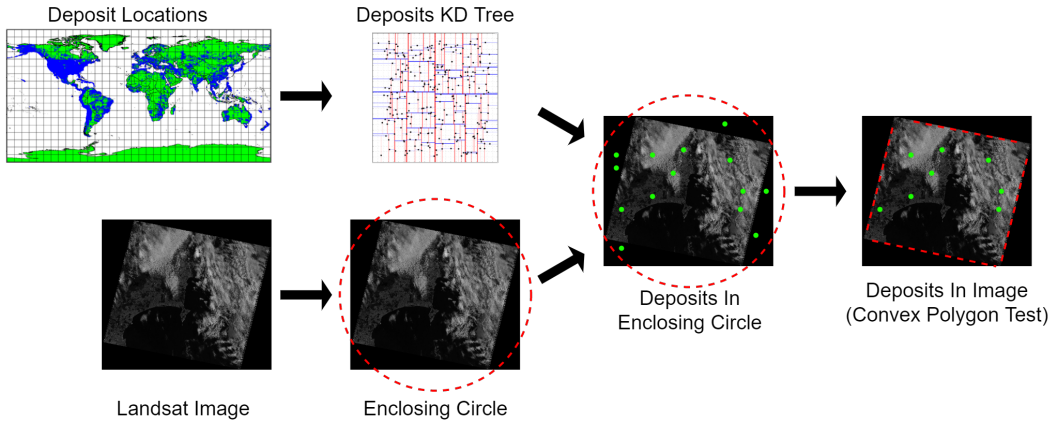


Figure 3: Label creation pipeline

Having obtained the set of deposits present in an image, separate labels were created for binary classification and regression. The binary classification labels indicated whether any minerals were present in the image, and the regression labels indicated the mineral richness of the image. The richness score was computed as the log transform of the absolute count of deposits in the image, normalized to the range 0-1. The log transfrom was used to reduce the skew of the regression values.

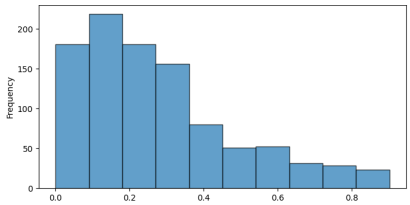


Figure 4: Distribution of log-normalised richness scores

$$label'_i = \frac{\ln(label_i + 1)}{max(labels)} \tag{1}$$

Location Buckets: Since Landsat images are taken at regular intervals, and our dataset contained several images at each location, it was necessary to ensure that no location appeared in both the training and testing sets simultaneously. To achieve this, the images were grouped by location, resulting in approximately 6000 unique location buckets.

Having grouped the images by location, the images in each bucket were sorted by average pixel brightness, which for band 7 images effectively sorts by degree of cloud cover. This permits the selection of training images having less cloud cover.

Finally, images in location buckets over the ocean were removed, to ensure that our models did not learn to simply detect land. The Python library Geopandas was used to load a geospatial dataset containing the polygons of the boundaries of all land on Earth; this dataset was then queried to determine whether each image's centerpoint was over land.

To prevent our models from simply learning the underlying distribution of training labels, we ensured that the ratio of class labels in the training set is balanced. In binary classficiation this is trivial, we enforce that the training set be comprised of 50% positive and 50% negative labels. In the regression case, the set of all examples has some underlying distribution shown in figure 4. The training set is then sampled randomly (without replacement) from this distribution, and therefore comes to have a very similar distribution. The remaining images are used as the validation set.

The training + testing set was drawn from the set of all images such that at least one image from each location bucket was included. The training set was then randomly sampled from this set, and the remaining images were used as the testing set. All experiments used a 80/20 split between training and testing sets.

Finally, since the Landsat images vary slightly in size and are not perfectly square, each image was padded to 512x512 pixels with black pixels, and all pixel brightnesses were normalized to the range 0-1.

2.2 Binary Classification

The binary classification problem setup was the first to be designed. It represents the simplest formulation of the problem, and our naive initial understanding. The primary goal of this problem was to determine whether a model trained on the dataset described above could learn anything relevant to the problem which would enable it to perform better than chance. To this end, examples with binary labels were prepared for the model as described above.

This approach initially proceded from the assumption that mineral deposits would be sparesely located throughout the world, and that their presence might correspond to certain easily detectable surface features. However, when the labelled location buckets were later plotted on a world map it became obvious that there was a clear preponderance of positive labels over land. Given the large surface area covered by each image almost all images contain at least some small number of desposits.

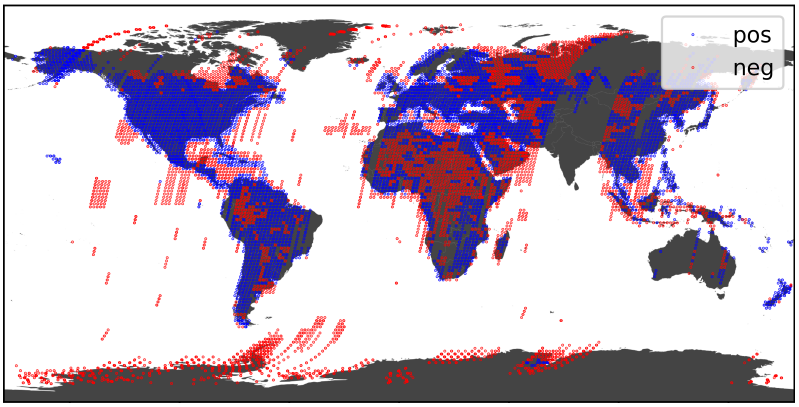


Figure 5: All labelled location buckets

Additionally, negatively labelled images do not necessarily mean that there are no mineral deposits in the region. In fact, this is a major philisophical problem with the problem setup. Many of the false

positives that the model will produce may in fact be true positives representing new desposits which the model has successfully identified. Without any way of independently verifying the presence of these deposits we are unable to truly evaluate the model's performance, and the model could never further learn from these examples.

Nevertheless, this problem formulation does allow us to successfully determine whether our dataset contains information which is learnable by a CNN.

2.3 Regression

The regression problem was designed to address the issues with the binary classification problem, and sought to investigate the feasibility of the practical application of our methods. In practice, a predicted richness score is more useful than a binary prediction when choosing a location to dig for minerals. The regression formulation also addresses the problem of whether negative examples are really negative: if a regression model achieves a low error when trained only on examples containing known deposits, then predicts a high richness score on an example containing no known deposits, it may indicate the presence of undiscovered deposits.

To improve the interpretability of our regression model performance, the training and testing sets for the regression problem were sampled such that each had a near-equivalent distribution of label values.

3 Background and Related Work

Much of the previous work done to integrate satellite and aerial photography into the mining industry has been to better assess geographic factors relevant as obstacles to construction and extraction [4]. As well as to predict the extent of damage down to the natural landscape, for instance, to determine if much tree cover would have to be removed. Other work at automatically detecting minerals from orbit has been limited by the sensor capabilities of the satellites, indeed typically only those sensors with very high resolution and particular imaging wavelengths are useful for geological applications.

4 Methods

4.1 Model Architecture

4.2 Hyperparameter Tuning

4.3 Model Evaluation

5 Results

5.1 Binary Classification

5.2 Regression

6 Discussion

6.1 Limitations

6.2 Future Work

7 Conclusion

References

[1] Landsat

[2] https://towardsdatascience.com/using-convolutional-neural-network-for-image-classification-5997bfd0ede4

[3] https://opendsa-server.cs.vt.edu/ODSA/Books/CS3/html/KDtree.html

[4] https://www.satimagingcorp.com/applications/energy/mining/