Soil Image Classification Part 1

Project Aim

This project focuses on identifying the type of soil from an image. Each image belongs to one of four types:

- Alluvial soil
- Black soil
- Clay soil
- Red soil

Our main goal is to build a model that can correctly classify all types of soil, not just the common ones. So, we're especially focusing on ensuring that even the rarest class is predicted accurately.

Dataset

We're given a set of labelled soil images for training and another set of unlabelled images for testing.

- The training images come with a CSV file that lists the soil type for each image.
- The test set includes image names only—we have to predict their soil types.
- All images are stored in separate folders for training and testing.

Cleaning and Preprocessing

Label Preparation

Before training the model, the soil type names are converted into numbers using label encoding. This helps the model process them as classification targets.

Dealing with Imbalance

Some soil types appear more often than others. To avoid the model becoming biased, we calculate weights for each class and adjust the training loss so that all soil types are treated fairly.

Image Cropping

Instead of using the full image (which might contain background noise), we apply a method called smart cropping. This looks for the main region in the image based on pixel intensity and crops around it, so the model focuses only on the actual soil part.

Transformations

We apply some tricks to help the model learn better:

- For training images, we slightly rotate, flip, and adjust colors randomly. This makes the model more flexible and less sensitive to changes.
- For validation and testing, we keep it simple, just resizing and normalizing.

The Model We Used

We chose ResNet-18, a well-known deep learning model that has been trained on millions of images before. This helps the model understand visual patterns better, even with a smaller dataset.

We replaced its last layer to output predictions for our 4 soil classes.

Some technical highlights:

- We used a method called transfer learning, starting with pretrained knowledge and fine-tuning it to our soil images.
- The loss function was modified to consider class imbalance.
- We used mixed precision training, which speeds things up and reduces GPU memory usage.

How We Trained the Model

We trained three different versions of the same model. Each time, we shuffled the training/validation split slightly to make the models learn a bit differently.

- For every training run, we trained for 15 rounds (epochs).
- After each round, we checked how well the model was doing especially looking at the worst performing class.
- If the model improved, we saved it. This way, we always kept the best version.

Making Predictions

Once all three models were trained, we used them together as a team.

- For each test image, we ran it through all three models.
- Each model gave its opinion (a probability for each class).
- We averaged these and picked the most confident class.

This kind of ensemble prediction helps smooth out mistakes from individual models and gives better final results.

Submission File

After prediction, we mapped the predicted numbers back to the original soil type names and saved everything in a submission.csv file. This file contains:

- image_id (the image name)
- soil_type (our predicted label)

This is the file that's submitted for evaluation.