# Problem

The goal of this project is to train a classifier with a small data set to identify if the leaf in an image is Coniferous or Deciduous.

The amount of data required to train a Deep Learning varies based on the project. 1,000 samples for each class are commonly considered as a minimum requirement. In this project, each class has only about 500 images. To get the best from the Deep Learning model, Transfer Learning method is applied to leverage the benefit of a pre-trained model.

The classifier, as the result of the project, can be further extended to more species. With more data and resources, it can be trained to recognize other leaves, trees, flowers that can be used by the customer to identify the plants they see.

The model can be trained to recognize the diseases of corps as well, which is useful for the farming and agriculture industry. With more specific data, it can be trained to determine the quality of the corps which can be used to automate the sorting tasks.

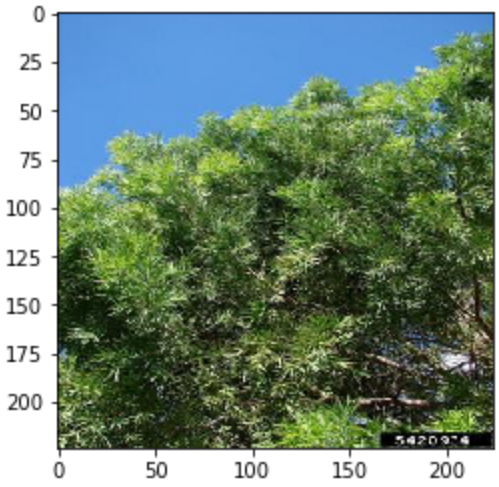
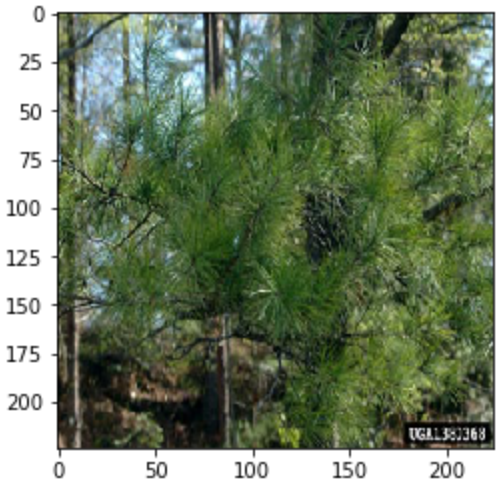
# Data

The images used for training and testing are scraped from Forestry Images (https://www.forestryimages.org/).

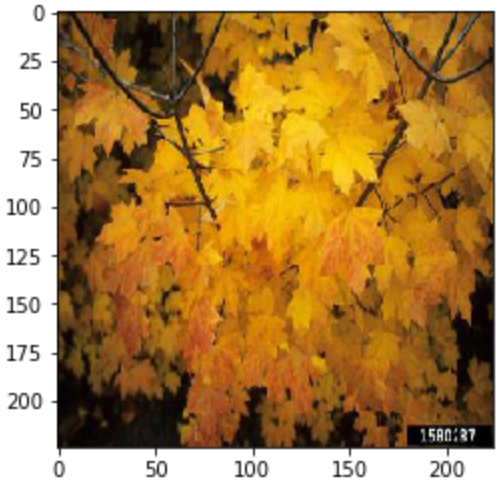
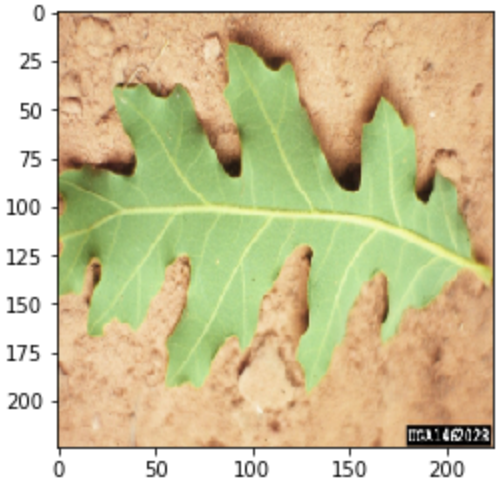
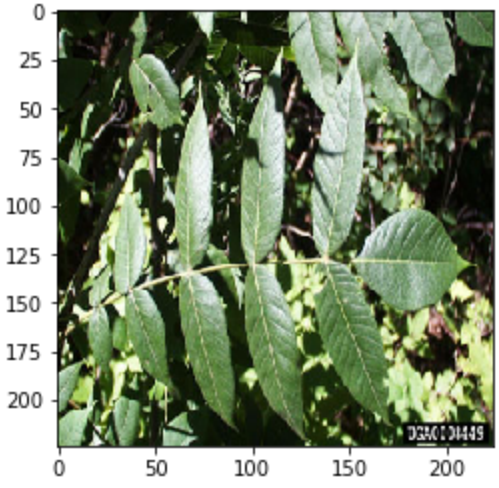
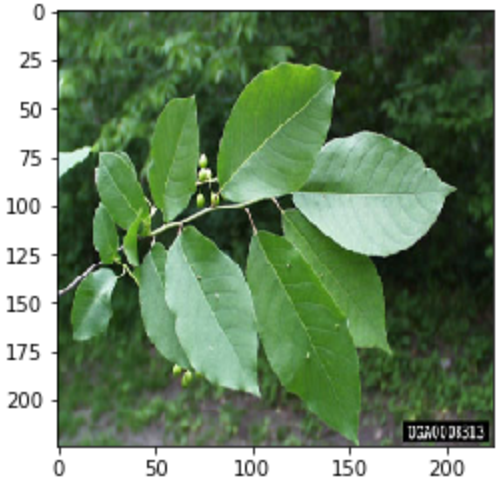
Compared with images from Google or other search engines, images from Forestry Images are less “noisy”. Images from Google or other search engines usually contain illustrations, drawings or misclassified images.

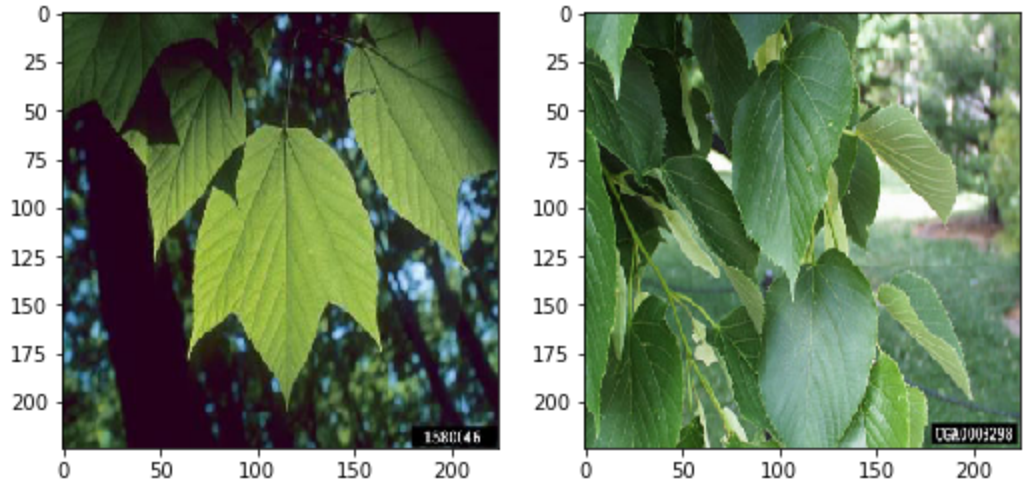
Forestry Images provide filter options that allow us to narrow the scope of the data. In this project, the classifier is trained with only Foliage (leaf) images. However, Cultivar and other kinds of images can be downloaded and used as training or validation data for the future.

**Coniferous Sample Images:**

**Deciduous Sample Images:**



# Model Training

## Technologies

### Transfer Learning

Transfer Learning has become a highly popular technique in developing Deep Learning models. It commonly has two steps:

1. Pretraining: The network is trained on a large scale benchmark dataset that has a wide diversity of classes.
2. Fine-tuning: The network is further trained on a smaller but more specific dataset that has fewer classes.

Transfer Learning technique is particularly good for this project in two reasons:

1. Small dataset: The project focuses on two classes and each class has 500 samples.
2. General features: The general features learned from the model designed for ImageNet with pre-trained weight can be reused for leaf classification.

### Keras

Keras is an open-source Neural Network library built on top of TensorFlow. It greatly reduces the repetitive work of constructing a Neural Network using TensorFlow. With good and easy to

use APIs, common use cases are handled with fewer user actions, which results in better and more productive development experience.

More importantly for this project, Keras has pre-trained applications that make transfer learning easy to implement. MobileNet model with pre-trained weights on ImageNet data is used here.

### MobileNet

MobileNet is a model built for mobile and embedded vision applications. MobileNet is chosen here because the best use case of the classifier trained in this project would be integration with mobile applications.

### ImageNet

ImageNet a large dataset designed for vision recognition research. It contains more than 20,000 classes. By using the applications in Keras, the resources required to pre-train the MobileNet model with ImageNet are greatly reduced.

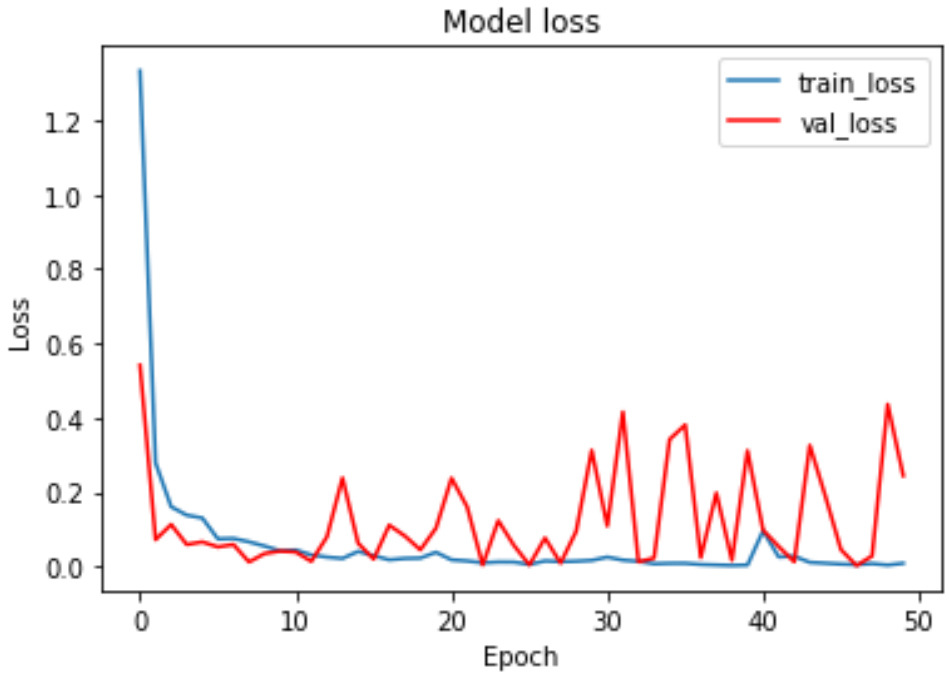
## Model Reconstruction

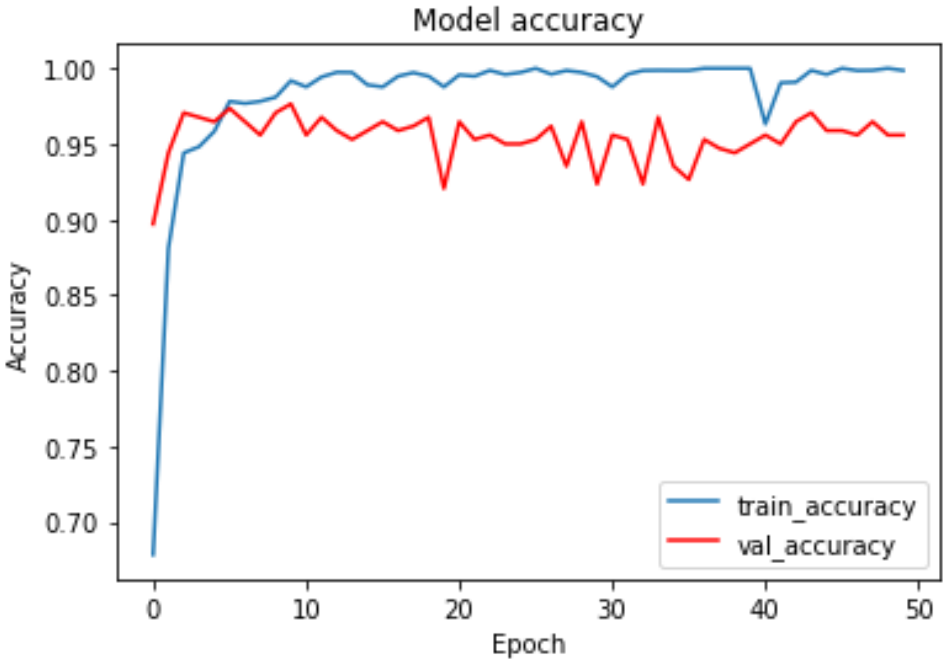
A 2D GlobalAveragePooling layer, a 512 Dense layer, and a Dropout layer are added to MobileNet model as the fine-tuning layers. The output layer is a Dense layer with 2 nodes for Coniferous and Deciduous classes.

`Adam` optimizer is used as the optimizer. `categorical_crossentropy` is used for loss function. `Accuracy` is the metric for evaluating the classifier.

## Training Result

30% of images are assigned as validation set. With 797 images belonging to 2 classes for training and 340 images for validation. The result is plotted as below:




The fluctuation of validation loss and accuracy is caused by the Dropout layer added to the model. The Dropout layer contains consists in randomly setting a fraction rate of input units to 0 at each update during training time, which helps prevent overfitting.

The average accuracy of validation is around 95%.

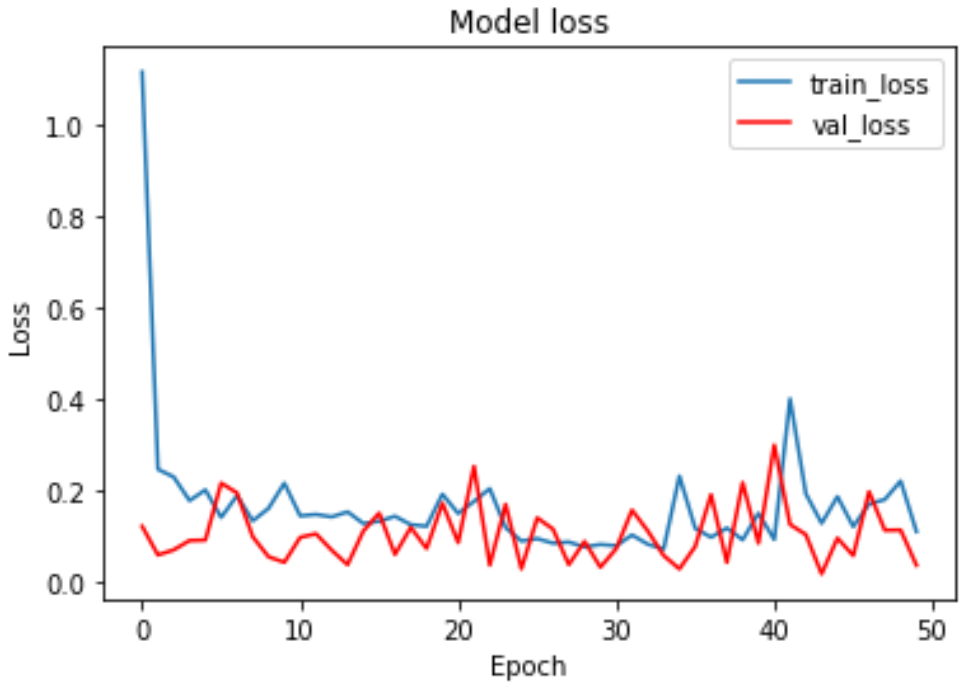
## Training Result With Image Augmentation

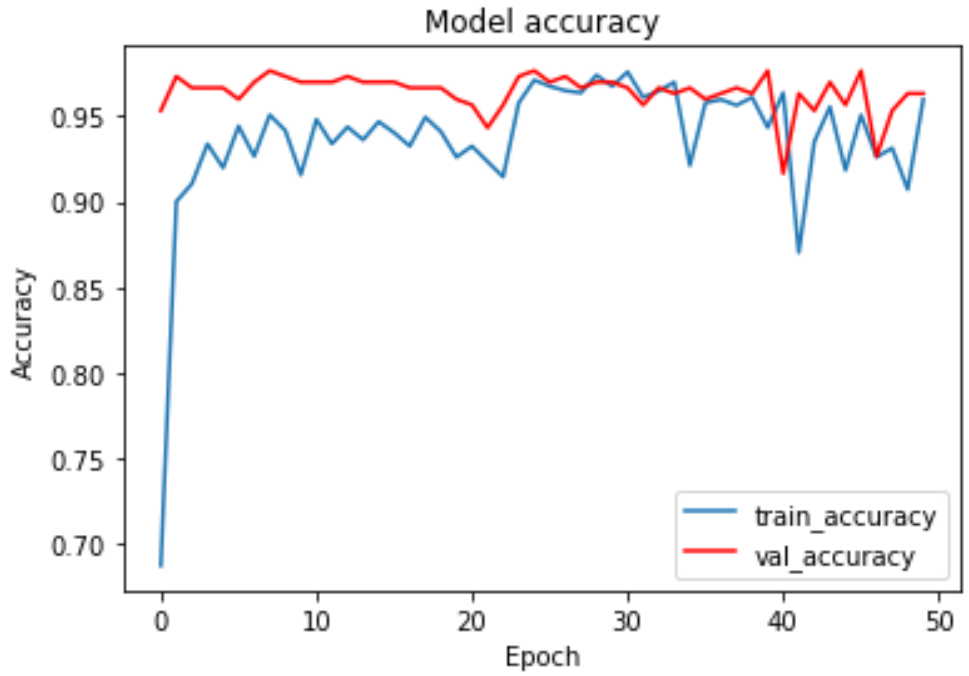
Augmentation is a common technique to expand the training data. By augmenting the images through a number of random transformations, the biggest obstacle for training the Deep Learning model can be partially resolved. Augmentation is only applied to training data as the validation data should be as close to real data as possible.

The following transformation operations are applied to the training data:

- Horizontal flip
- Vertical flip
- Rotation
- Brightness modification
- Zoom in/out

The new result with augmented images:



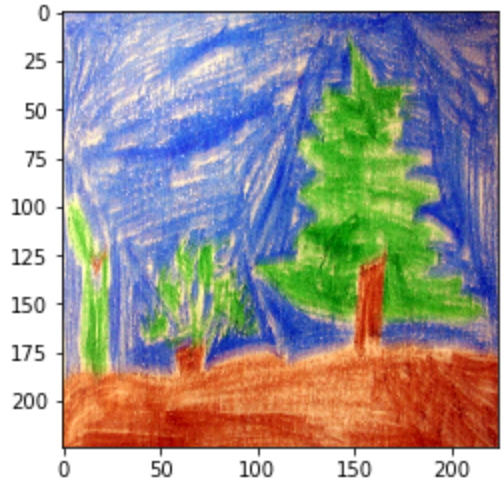



The validation loss and accuracy represent a more stable state with augmented training data. The average accuracy of validation increases to 97%.

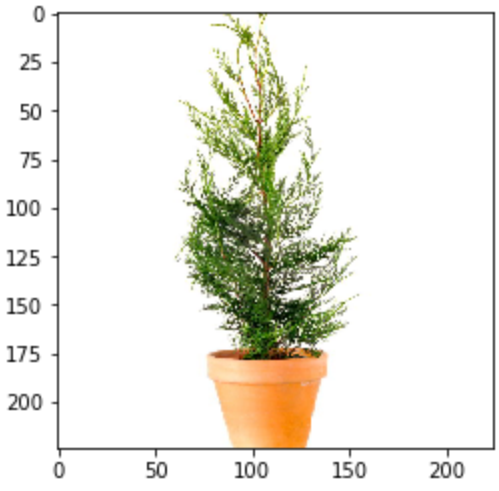
## Beyond Leaf Images

We are not satisfied by testing with only leaf images. How does the model perform on other images like Coniferous and Deciduous trees?

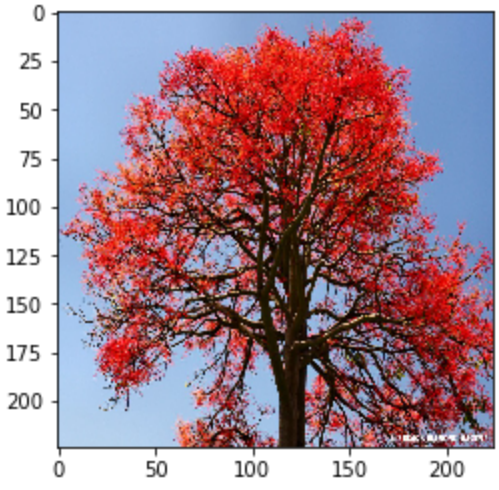
Here is a bit of the exploration:



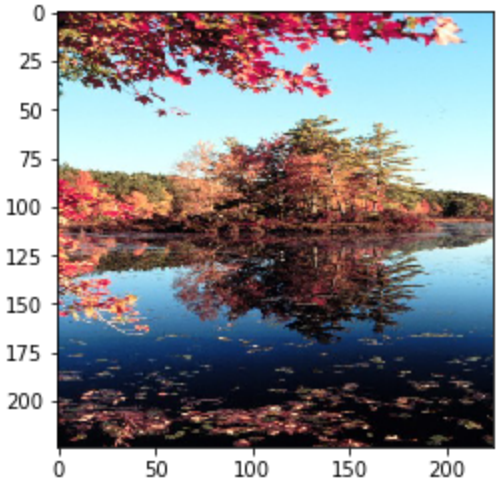

{'coniferous': 0.877267, 'deciduous': 0.122732975}



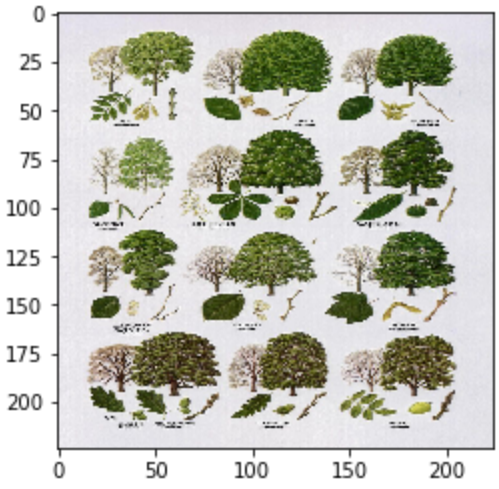

{'coniferous': 0.9939142, 'deciduous': 0.0060858116}



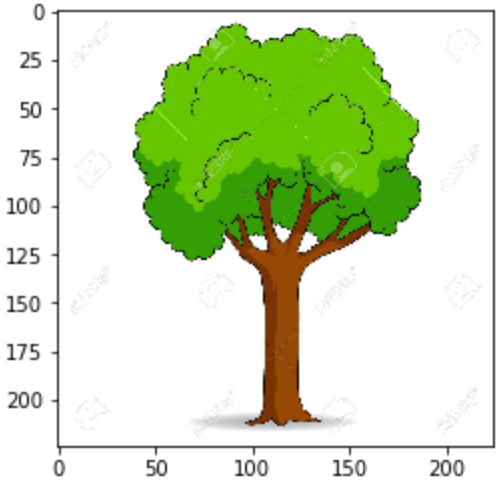

{'coniferous': 0.9517209, 'deciduous': 0.048279036}



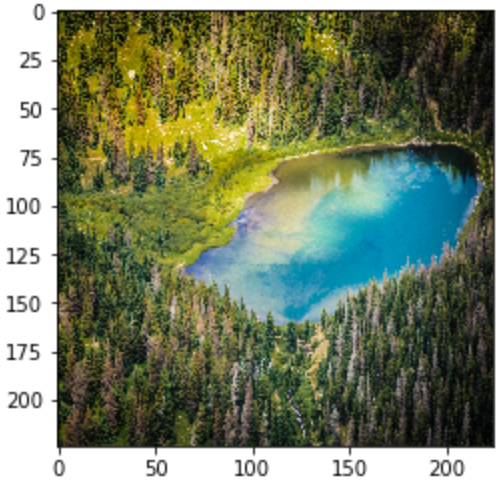

{'coniferous': 0.70984715, 'deciduous': 0.29015285}



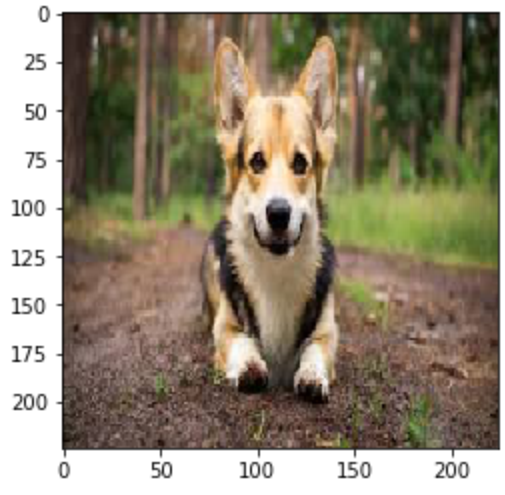

{'coniferous': 0.78804475, 'deciduous': 0.21195523}




{'coniferous': 0.054971006, 'deciduous': 0.94502896}




{'coniferous': 0.6697901, 'deciduous': 0.33020994}



{'coniferous': 0.47172523, 'deciduous': 0.52827483}

The model performs pretty for tree images or even drawings. Please ignore the last dog image.

# Next

There are a few directions and tons of possibilities for the next step of this project.

## Specific Data For Specific Applications

Trained with specific data, the model can be integrated with various applications from plant identification, disease recognition, corp sorting, etc.

## Understand More About What The Model Have Learned

Patterns learned from the model can be output and displayed, which helps understand and improve how the model work.

## Different Model Architecture and Pre-trained Weight

MobileNet model architecture with ImageNet weights is used for the project. However, there are many other architectures and pre-trained weights available. Some of them may have better performance on the problem this project tries to solve.