CS C200A Final Project: Image Classification

Zetian Xiao, Sikai Yin

Abstract

In this final project, we did image classification task based on the given dataset. We went through the Data Science life cycle for this task over the given image dataset. The whole project was divided into five parts, including data input, exploratory data analysis and feature extraction, classifier training, classifier performance assessment and neural networks model. Through these different steps, we comprehensively reviewed and practiced lessons we learned in the course in this semester.

Introduction

With the development in Data Science and Machine Learning techniques in recent years, we have seen more and more successful applications based on Artificial Intelligence. As the fundamental task in Computer Vision and Pattern Recognition, image classification has long been a hot research field. In image classification workflow, we need to develop the model that could classify images based on certain image features. In order to achieve this workflow, traditional classification models require manually extract features from image data, which requires a lot of image processing and feature selection work. Nowadays, thanks to neural network models, we will be able to train large-scale image classifications that automatically select useful image features. As the fundamental Data Science course, we learned techniques and applications across the Data Science life cycle in this semester, and this project provides us a good opportunity to go through this process and compare the traditional classification model with neural network classifiers.

Data Description

In this project, we are given a learning set consisting of 2021 images from 20 different types. The given images have different sizes and each of them is represented as the 3-d array with RGB format. Each value in the array has color intensity between 0 and 2^8-1. The 20 types and the number of images for each type are described in the following table.

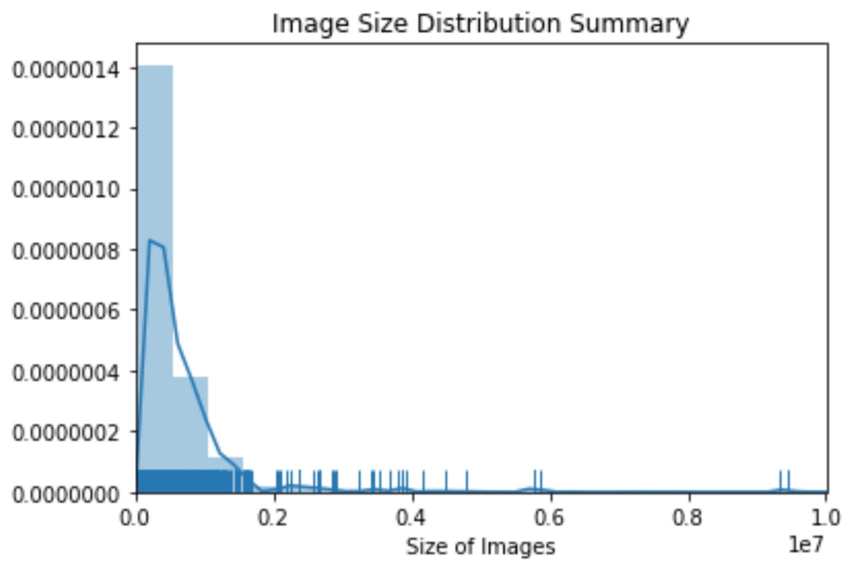
Exploratory Data Analysis

With the given dataset, we conducted the Exploratory Data Analysis(EDA) in the following aspects.

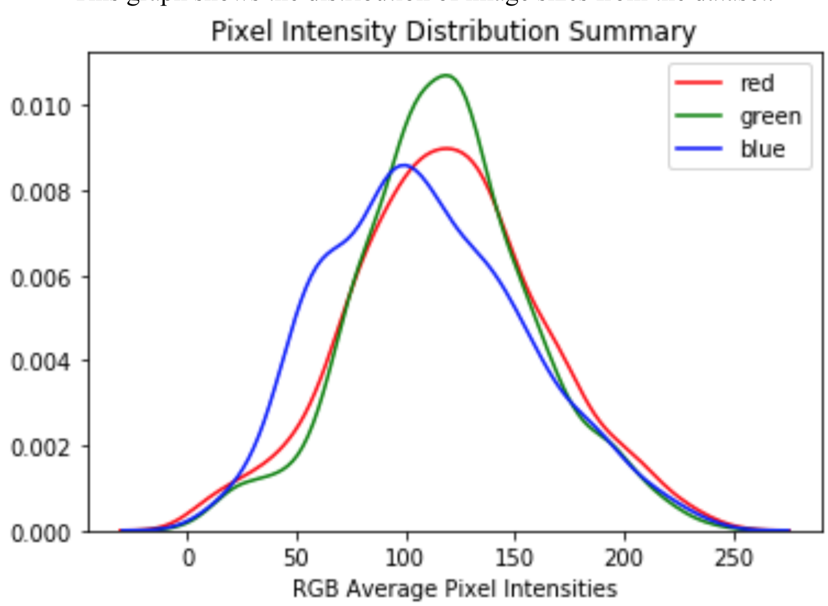
a.Data Cleaning and Transformation:

1. Because some images only have 1 channel, we manually increase the number of channel to 3(using OpenCV grey to RGB)
2. Different plots have different sizes but Convolutional Neural Network requires plots with equal size. Therefore, before we train the Convolutional Neural Network we resize our image to 224*224.

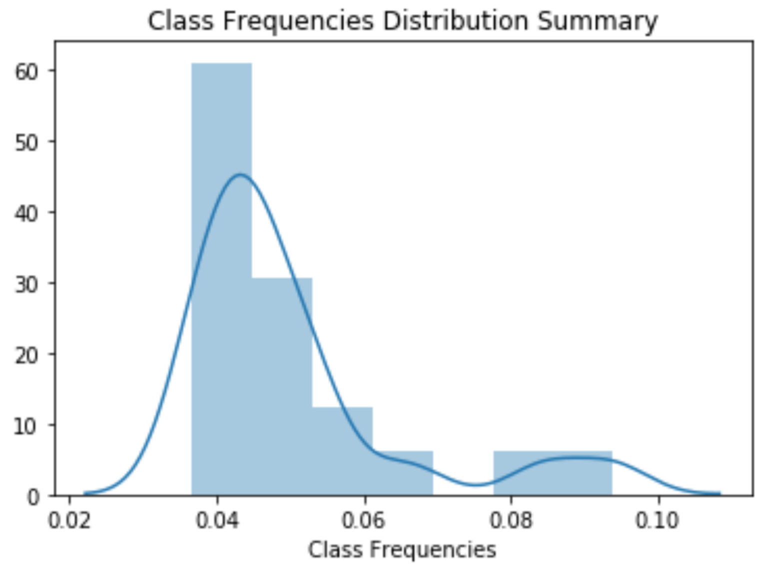
b.Plot for image size, pixel, class count distribution



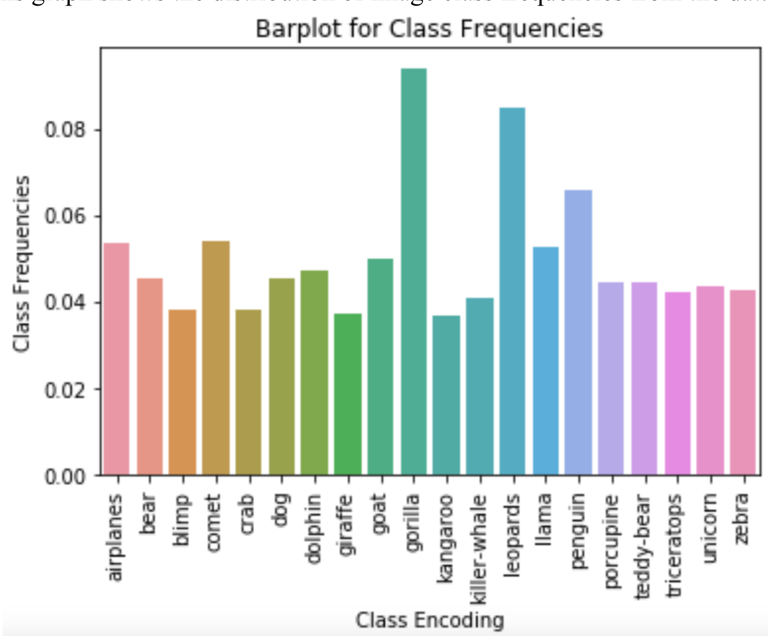
This graph shows the distribution of image sizes from the dataset.



This graph shows the distribution of RGB pixel intensities from the dataset



This graph shows the distribution of image class frequencies from the dataset



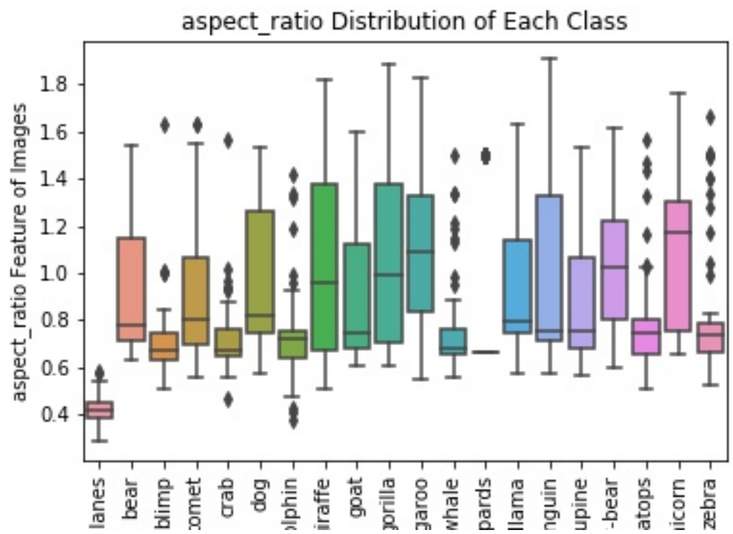
This barplot shows the relative frequencies of each class from the dataset

c. Feature List

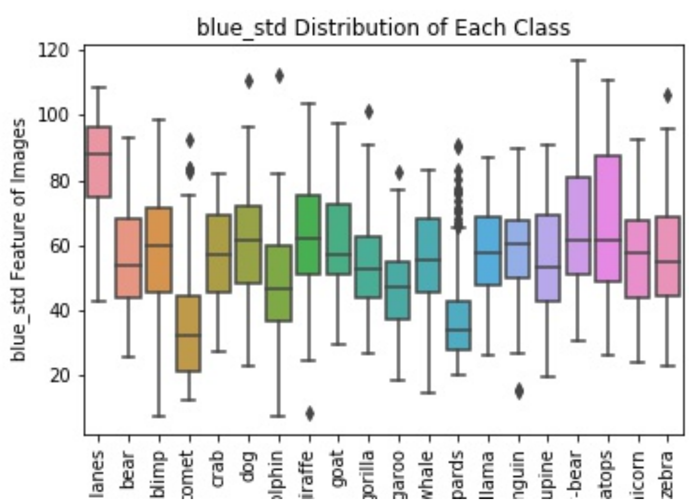
1. Image size
2. Average intensities of the red, green and blue channel
3. Standard deviation of the red, green and blue channel
4. SIFT: SIFT algorithm is used to find the key point in the plot. We use the length of detected keypoints by SIFT algorithm as our image feature
5. Aspect Ratio of image size
6. Hog Feature: We use the histogram of Oriented Gradients(HOG) computed from the image as a feature, here we set the length of Hog Feature to be 16
7. Color Histogram: We use the histogram of the image as the image feature, to limit its size, we set the bins of color histogram to be 3.

8. Image Hog Standard Deviation Value: We use the std value of Oriented Gradients(HOG) computed from the image as a feature.
9. Image Hog Mean Value: We use the mean value of Oriented Gradients(HOG) computed from the image as a feature.
10. Local Binary Pattern Feature: Local binary patterns(LBP) is a type of visual descriptor used for image classification. Here we compute the histogram from Local binary patterns(LBP) to use as the feature.

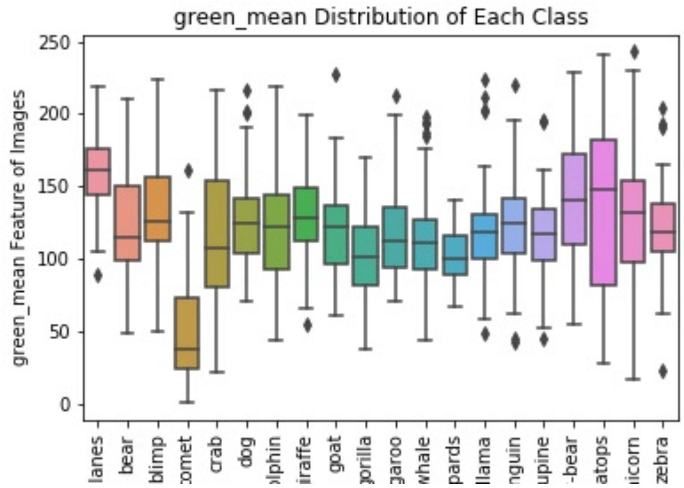
d.Variations of Image feature between classes



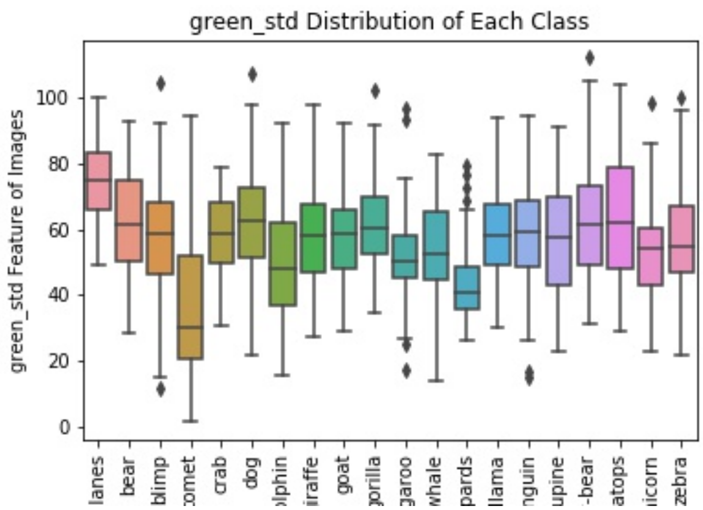
This boxplot shows the variation of image aspect ratio feature between different classes



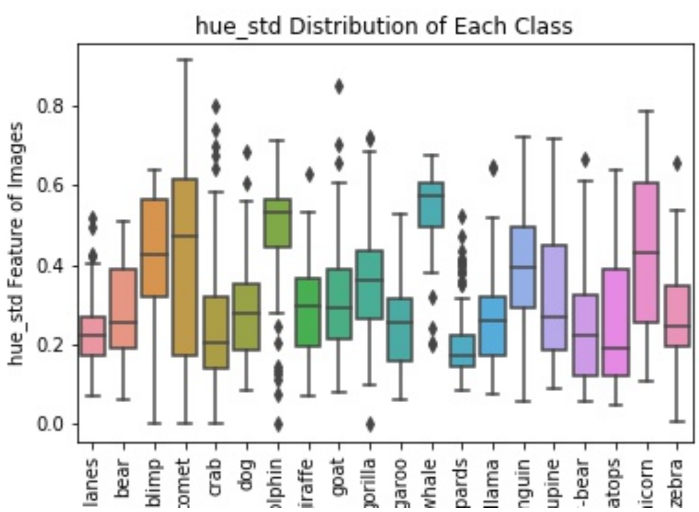
This boxplot shows the variation of image standard derivation of Blue channel feature between different classes



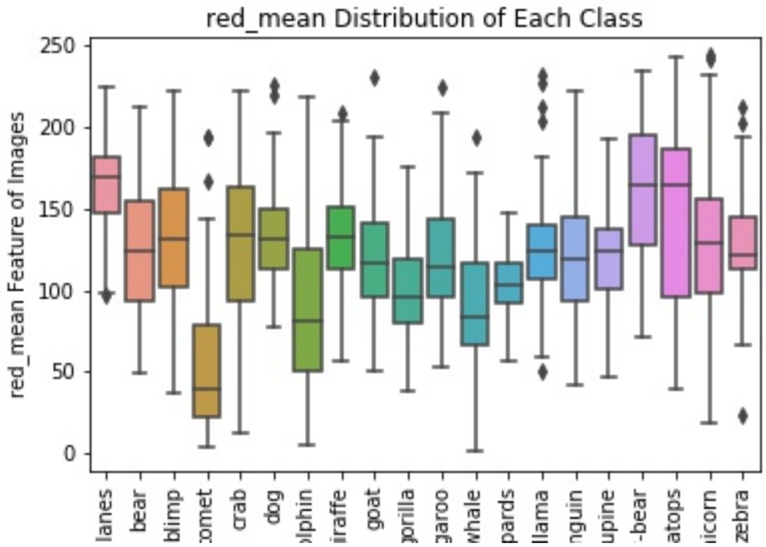
This boxplot shows the variation of image mean value of Green channel feature between different classes



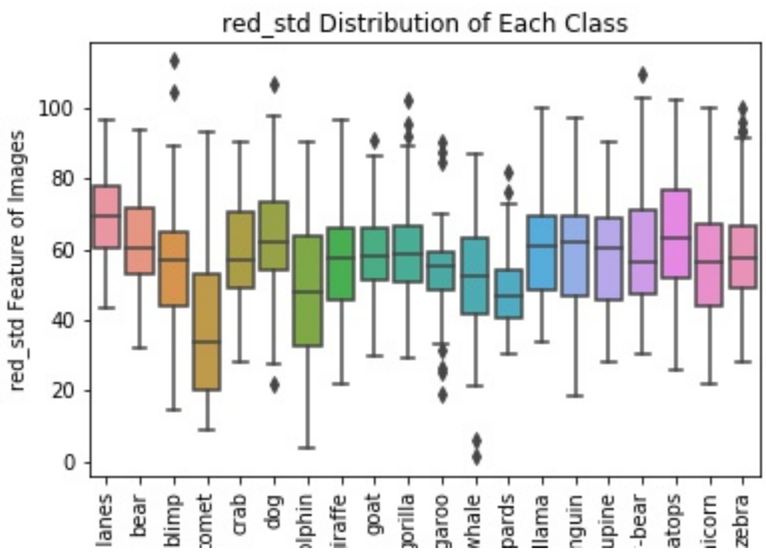
This boxplot shows the variation of image standard deviation of Green channel feature between different classes



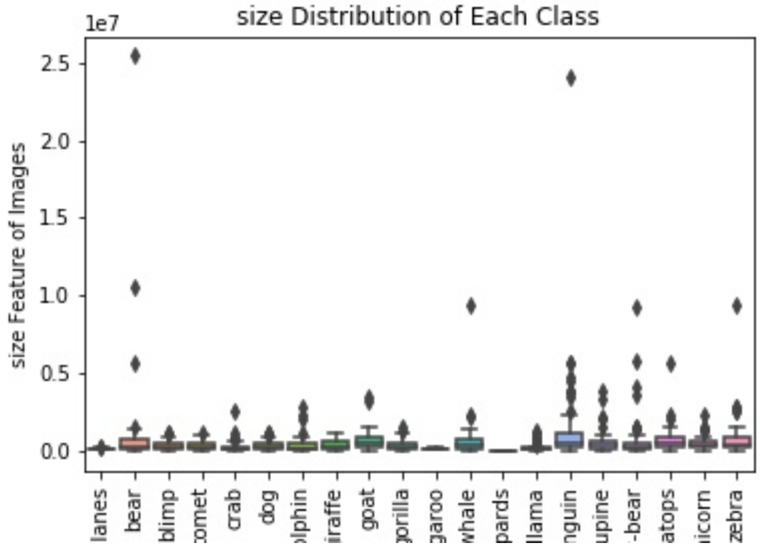
This boxplot shows the variation of image Hue mean value feature between different classes



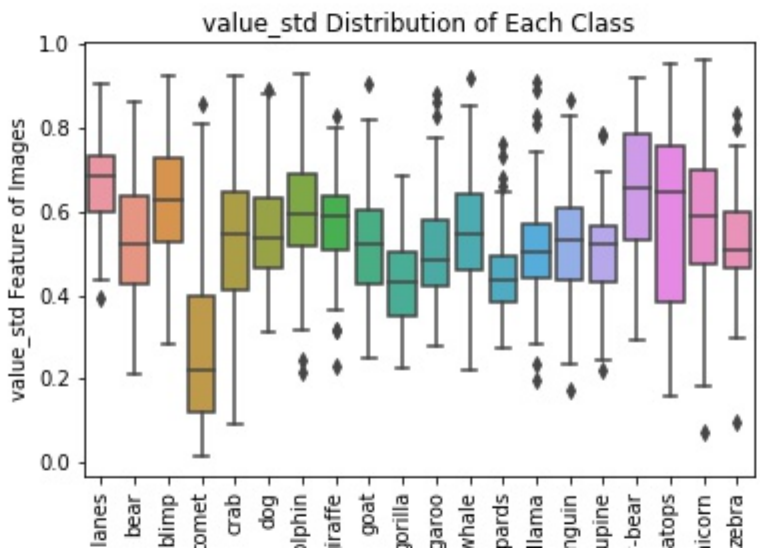
This boxplot shows the variation of image mean value of Red channel feature between different classes



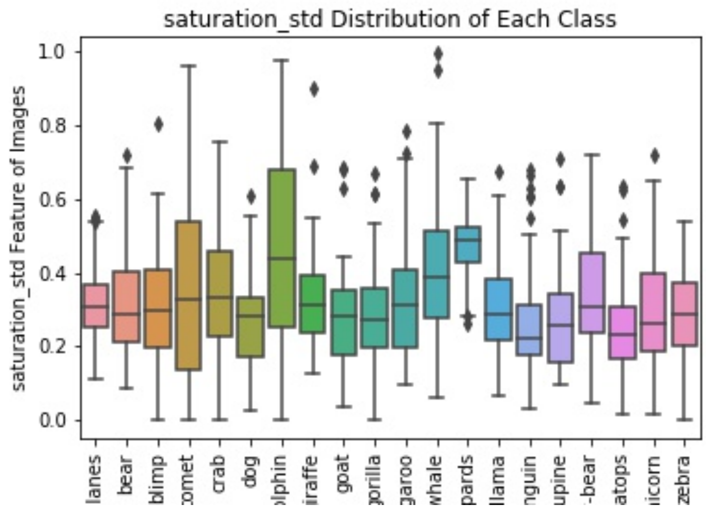
This boxplot shows the variation of image standard deviation of Red channel feature between different classes



This boxplot shows the variation of image size between different classes



This boxplot shows the variation of image Value mean value feature between different classes



This boxplot shows the variation of image Saturation mean value feature between different classes

e.Feature Selection
In our model training, we find some of over features are redundant and would lead to the colinear problems and slow the training speed. To avoid these problems, we excluded some of the features in our training set, like hog_mean, hog_std and SUFT features. The model training after this selection was greatly accelerated.

Traditional Model Training:

We use 5 fold cross-validation to test the performance of different models. Below is the result for our Cross-Validation result

Model Name	Mean Accuracy Score on 5-Fold Cross Validation
Logistic Regression	0.4337209302325581
K-Nearest Neighbors	0.19988039867109636
Classification Tree	0.2951339977851606
Random Forest	0.35311627906976745
Support Vector Machine	0.05130675526024363

We find that logistic regression has the best accuracy. Therefore, we decide to use logistic regression as our final model.

Confusion Matrix:
From the confusion matrix (We did not put the confusing matrix here because the table size is 20*20 which is too big), we can see that for our logistic regression model, most bears are classified as the

gorilla, and no dog is correctly classified. Also, it seems that zebra is a quite challenging class that LR model misclassify much input as zebra. Our model achieves 100% classification accuracy on leopards class.

Neural Network:

Because this is an image dataset, we decided to use the Convolutional Neural Network(CNN) to solve this problem with potentially higher accuracy. We use PyTorch to created our model. Initially, we created CNN with 2 convolutional layers and two fully connected layers, but the model overfits very quickly because the number of training samples is too small. The training accuracy keeps increasing but the validation accuracy remains around 30%. Therefore, we added data augmentation in our data transformation steps. Each time an image was passed to train, it would be added some jitter pixels, cropped at random position, rotate with random angles, and flipped with certain probabilities. After doing image augmentation, I added the Dropout layer for our network to further prevent overfitting. After these steps, our model validation accuracy could reach 50%. We add additional convolutional layers and pooling layer for our network to increase complexity, and the validation accuracy now could reach 60%.

Future improvement:

For the traditional method, we did not tune the parameters enough. In the future, it is better to use tuning techniques for example, grid search to reach better accuracy
For the Neural network, we need more time tunning parameters, for example, kernel size, padding, stride, pooling size, etc. in order to reach higher accuracy